**文件编号：**

MS-002P

软件详细设计说明书

编制/日期：

审核/日期：

批准/日期：

杭州三坛医疗科技有限公司

文档更改履历

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 发布/实施日期 | 更改内容概述 | 更改者 |
| V1.0 | 2022.07.26 | 文件新编 | 高广文 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目 录**

[1. 引言 1](#_Toc27318)

[1.1. 编写目的 1](#_Toc13757)

[1.2. 项目背景 1](#_Toc4421)

[1.3. 术语及缩写词 1](#_Toc7877)

[1.4. 参考资料 2](#_Toc21545)

[2. 总体设计 3](#_Toc8133)

[2.1. 处理流程 3](#_Toc10569)

[2.2. 总体结构和模块外部设计 8](#_Toc30887)

[3. 启动模块 9](#_Toc29697)

[3.1. 模块描述 9](#_Toc32264)

[3.2. 功能 9](#_Toc609)

[3.3. 性能 9](#_Toc2554)

[3.4. 输入项目 10](#_Toc367)

[3.5. 输出项目 10](#_Toc19391)

[3.6. 算法 10](#_Toc2160)

[3.7. 程序逻辑 11](#_Toc13931)

[3.8. 接口 12](#_Toc7043)

[3.9. 存储分配 14](#_Toc3473)

[3.10. 限制条件 14](#_Toc18453)

[3.11. 测试要点 14](#_Toc18538)

[4. 登录模块 14](#_Toc20986)

[4.1. 模块描述 14](#_Toc1810)

[4.2. 功能 15](#_Toc24099)

[4.3. 性能 15](#_Toc29393)

[4.4. 输入项目 15](#_Toc12375)

[4.5. 输出项目 15](#_Toc27018)

[4.6. 算法 15](#_Toc31570)

[4.7. 程序逻辑 16](#_Toc5574)

[4.8. 接口 17](#_Toc32565)

[4.9. 存储分配 17](#_Toc24567)

[4.10. 限制条件 18](#_Toc20532)

[4.11. 测试要点 18](#_Toc19170)

[5. 患者管理模块 18](#_Toc12251)

[5.1. 模块描述 18](#_Toc7253)

[5.2. 功能 18](#_Toc30058)

[5.3. 性能 19](#_Toc31891)

[5.4. 输入项目 19](#_Toc9745)

[5.5. 输出项目 19](#_Toc18972)

[5.6. 算法 20](#_Toc9239)

[5.7. 程序逻辑 21](#_Toc3262)

[5.8. 接口 22](#_Toc21554)

[5.9. 存储分配 24](#_Toc2571)

[5.10. 限制条件 25](#_Toc21141)

[5.11. 测试要点 25](#_Toc20137)

[6. 方案与流程管理模块 25](#_Toc23939)

[6.1. 模块描述 25](#_Toc23332)

[6.2. 功能 25](#_Toc2407)

[6.3. 性能 26](#_Toc22553)

[6.4. 输入项目 26](#_Toc12995)

[6.5. 输出项目 26](#_Toc5368)

[6.6. 算法 26](#_Toc1135)

[6.7. 程序逻辑 26](#_Toc28087)

[6.8. 接口 27](#_Toc23811)

[6.9. 存储分配 29](#_Toc22308)

[6.10. 限制条件 29](#_Toc29793)

[6.11. 测试要点 29](#_Toc3768)

[7. CT导入模块 29](#_Toc27320)

[7.1. 模块描述 29](#_Toc4636)

[7.2. 功能 30](#_Toc29816)

[7.3. 性能 31](#_Toc25533)

[7.4. 输入项目 31](#_Toc13071)

[7.5. 输出项目 31](#_Toc3367)

[7.6. 算法 31](#_Toc2152)

[7.7. 程序逻辑 31](#_Toc25472)

[7.8. 接口 33](#_Toc3499)

[7.9. 存储分配 35](#_Toc13971)

[7.10. 限制条件 35](#_Toc19345)

[7.11. 测试要点 35](#_Toc2982)

[8. 四视图模块 35](#_Toc939)

[8.1. 模块描述 35](#_Toc4735)

[8.2. 功能 35](#_Toc25160)

[8.3. 性能 36](#_Toc6990)

[8.4. 输入项目 36](#_Toc24462)

[8.5. 输出项目 36](#_Toc19336)

[8.6. 算法 36](#_Toc2395)

[8.7. 程序逻辑 36](#_Toc1742)

[8.8. 接口 37](#_Toc29986)

[8.9. 存储分配 39](#_Toc5074)

[8.10. 限制条件 39](#_Toc25002)

[8.11. 测试要点 39](#_Toc1432)

[9. 规划模块 39](#_Toc4064)

[9.1. 模块描述 39](#_Toc29570)

[9.2. 功能 40](#_Toc21273)

[9.3. 性能 41](#_Toc7950)

[9.4. 输出项目 42](#_Toc7282)

[9.5. 算法 42](#_Toc23598)

[9.6. 程序逻辑 42](#_Toc8758)

[9.7. 接口 45](#_Toc31128)

[9.8. 存储分配 48](#_Toc2930)

[9.9. 限制条件 48](#_Toc4241)

[9.10. 测试要点 48](#_Toc2332)

[10. 术前准备模块 49](#_Toc8180)

[10.1. 模块描述 49](#_Toc20853)

[10.2. 功能 49](#_Toc13444)

[10.3. 性能 49](#_Toc22423)

[10.4. 输入项目 49](#_Toc12494)

[10.5. 输出项目 50](#_Toc10312)

[10.6. 算法 50](#_Toc6442)

[10.7. 程序逻辑 50](#_Toc24200)

[10.8. 接口 50](#_Toc11409)

[10.9. 存储分配 52](#_Toc30073)

[10.10. 限制条件 52](#_Toc6506)

[10.11. 测试要点 52](#_Toc10891)

[11. X片注册模块 52](#_Toc21884)

[11.1. 模块描述 52](#_Toc31257)

[11.2. 功能 53](#_Toc6818)

[11.3. 性能 54](#_Toc20952)

[11.4. 输入项目 54](#_Toc22008)

[11.5. 输出项目 54](#_Toc12225)

[11.6. 算法 55](#_Toc1433)

[11.7. 程序逻辑 57](#_Toc22887)

[11.8. 接口 60](#_Toc8489)

[11.9. 存储分配 65](#_Toc30419)

[11.10. 限制条件 65](#_Toc19382)

[11.11. 测试要点 65](#_Toc30040)

[12. 图像配准模块 65](#_Toc16835)

[15.1. 模块描述 65](#_Toc31234)

[15.2. 功能 65](#_Toc19568)

[15.2. 性能 66](#_Toc10385)

[15.3. 输入项目 66](#_Toc2847)

[15.4. 输出项目 67](#_Toc27413)

[15.5. 算法 67](#_Toc12038)

[15.7. 程序逻辑 67](#_Toc1961)

[15.8. 接口 70](#_Toc24718)

[15.9. 存储分配 77](#_Toc31975)

[15.10. 限制条件 78](#_Toc1136)

[15.11. 测试要点 78](#_Toc8965)

[13. 定位导航模块 78](#_Toc1773)

[13.1. 模块描述 78](#_Toc16051)

[13.2. 功能 79](#_Toc17813)

[13.3. 性能 81](#_Toc23616)

[13.4. 输入项目 81](#_Toc13433)

[13.5. 输出项目 81](#_Toc15502)

[13.6. 算法 81](#_Toc446)

[13.7. 程序逻辑 82](#_Toc10960)

[13.8. 接口 83](#_Toc21221)

[13.8.1. ArmSimulationWidget 83](#_Toc2357)

[13.8.2. ScrewTableView 85](#_Toc20811)

[13.8.3. Location2DWidget 86](#_Toc5095)

[13.9. 存储分配 89](#_Toc10154)

[13.10. 限制条件 89](#_Toc5385)

[13.11. 测试要点 90](#_Toc24741)

[14. 反馈验证模块 90](#_Toc22936)

[14.1. 模块描述 90](#_Toc22175)

[14.2. 功能 90](#_Toc30296)

[14.3. 性能 91](#_Toc10099)

[14.4. 输入项目 91](#_Toc9531)

[14.5. 输出项目 91](#_Toc29385)

[14.6. 算法 92](#_Toc1758)

[14.7. 程序逻辑 92](#_Toc8209)

[14.8. 接口 93](#_Toc12562)

[14.9. 存储分配 95](#_Toc4468)

[14.10. 限制条件 95](#_Toc31372)

[14.11. 测试要点 95](#_Toc4925)

[15. 配置管理模块 96](#_Toc13613)

[15.1. 模块描述 96](#_Toc4994)

[15.2. 功能 96](#_Toc21384)

[15.3. 性能 97](#_Toc19476)

[15.4. 输入项目 98](#_Toc7050)

[15.5. 输出项目 98](#_Toc6404)

[15.6. 算法 98](#_Toc22388)

[15.7. 程序逻辑 98](#_Toc18494)

[15.8. 接口 99](#_Toc16257)

[15.9. 存储分配 101](#_Toc21255)

[15.10. 限制条件 101](#_Toc27123)

[15.11. 测试要点 102](#_Toc14437)

[16. 设备自检模块 102](#_Toc15455)

[16.1. 模块描述 102](#_Toc10160)

[16.2. 功能 102](#_Toc28777)

[16.3. 性能 103](#_Toc25835)

[16.4. 输入项目 103](#_Toc3922)

[16.5. 输出项目 103](#_Toc23692)

[16.6. 算法 104](#_Toc8963)

[16.7. 程序逻辑 104](#_Toc19164)

[16.8. 接口 105](#_Toc4229)

[16.9. 存储分配 106](#_Toc15137)

[16.10. 限制条件 106](#_Toc28408)

[16.11. 测试要点 106](#_Toc1313)

[17. 日志管理模块 106](#_Toc1931)

[17.1. 模块描述 106](#_Toc25545)

[17.2. 功能 106](#_Toc19982)

[17.3. 性能 107](#_Toc22007)

[17.4. 输入项目 107](#_Toc24321)

[17.5. 输出项目 107](#_Toc17088)

[17.6. 算法 107](#_Toc14175)

[17.7. 程序逻辑 107](#_Toc25760)

[17.8. 接口 108](#_Toc15215)

[17.9. 存储分配 108](#_Toc23419)

[17.10. 限制条件 109](#_Toc3421)

[17.11. 测试要点 109](#_Toc856)

[18. 双目相机模块 109](#_Toc31088)

[18.1. 模块描述 109](#_Toc24123)

[18.2. 功能 109](#_Toc11540)

[18.3. 性能 109](#_Toc4491)

[18.4. 输入项目 110](#_Toc22291)

[18.5. 输出项目 110](#_Toc15331)

[18.6. 算法 110](#_Toc687)

[18.7. 程序逻辑 110](#_Toc20037)

[18.8. 接口 111](#_Toc19584)

[18.8.1.NDIMapping 112](#_Toc11270)

[18.9. 存储分配 113](#_Toc18238)

[18.10. 限制条件 113](#_Toc12010)

[18.11. 测试要点 113](#_Toc22495)

[19. 机械臂控制模块 113](#_Toc22853)

[19.1. 模块描述 113](#_Toc3623)

[19.2. 功能 113](#_Toc16501)

[19.3. 性能 114](#_Toc10200)

[19.4. 输入项目 114](#_Toc30009)

[19.5. 输出项目 114](#_Toc18216)

[19.6. 算法 114](#_Toc5392)

[19.7. 程序逻辑 115](#_Toc19050)

[19.8. 接口 115](#_Toc20120)

[19.8.1. RobtParaAdjustWidget 115](#_Toc4364)

[19.8.2. RobotSimulationWidget 116](#_Toc27046)

[19.8.3. RobotArmViewWidget 117](#_Toc21271)

[19.9. 存储分配 120](#_Toc11926)

[19.10. 限制条件 120](#_Toc6201)

[19.11. 测试要点 120](#_Toc24351)

[20. C臂机模块 120](#_Toc22128)

[20.1. 模块描述 120](#_Toc17957)

[20.2. 功能 120](#_Toc2301)

[20.3. 性能 121](#_Toc4539)

[20.4. 输入项目 121](#_Toc11802)

[20.5. 输出项目 121](#_Toc30182)

[20.6. 算法 121](#_Toc17853)

[20.7. 程序逻辑 121](#_Toc9908)

[20.8. 接口 122](#_Toc32714)

[20.9. 存储分配 123](#_Toc30792)

[20.10. 限制条件 123](#_Toc23675)

[20.11. 测试要点 123](#_Toc31759)

[21. 系统功能模块 123](#_Toc22843)

[21.1. 模块描述 123](#_Toc27824)

[21.2. 功能 123](#_Toc21847)

[21.3. 性能 124](#_Toc1662)

[21.4. 输入项目 124](#_Toc29995)

[21.5. 输出项目 124](#_Toc6255)

[21.6. 算法 125](#_Toc16306)

[21.7. 程序逻辑 125](#_Toc14827)

[21.8. 接口 125](#_Toc20902)

[21.9. 存储分配 126](#_Toc13297)

[21.10. 限制条件 126](#_Toc6802)

[21.11. 测试要点 126](#_Toc24661)

[22. 其他要求 127](#_Toc31622)

# 引言

## 编写目的

本详细设计说明书是为了规范及保证项目工作合理有序的开展，经概要设计评审后进一步细化，分别对每一模块进行详细细化包括解决方案、接口等方面的设计，明确描述所有输入输出参数、类型逻辑算法以及调用关系。作为开发人员和测试人员进一步变成和编写测试用例依据。

本项目详细设计说明书用于从具体上指导骨科手术规划与控制软件的开发，并面向项目组全体成员。

## 项目背景

传统的骨科手术非常依赖有经验的医生，手术过程也较复杂且漫长。为了能更加精准的定位病灶，缩短手术的时间，并尽可能减轻患者的痛苦，现开发骨科手术规划与控制软件，作为骨科手术导航定位系统的重要组成部分，主要用于手术规划通道，控制机械臂定位并通过双目红外相机实时监测，指导外科手术。

## 术语及缩写词

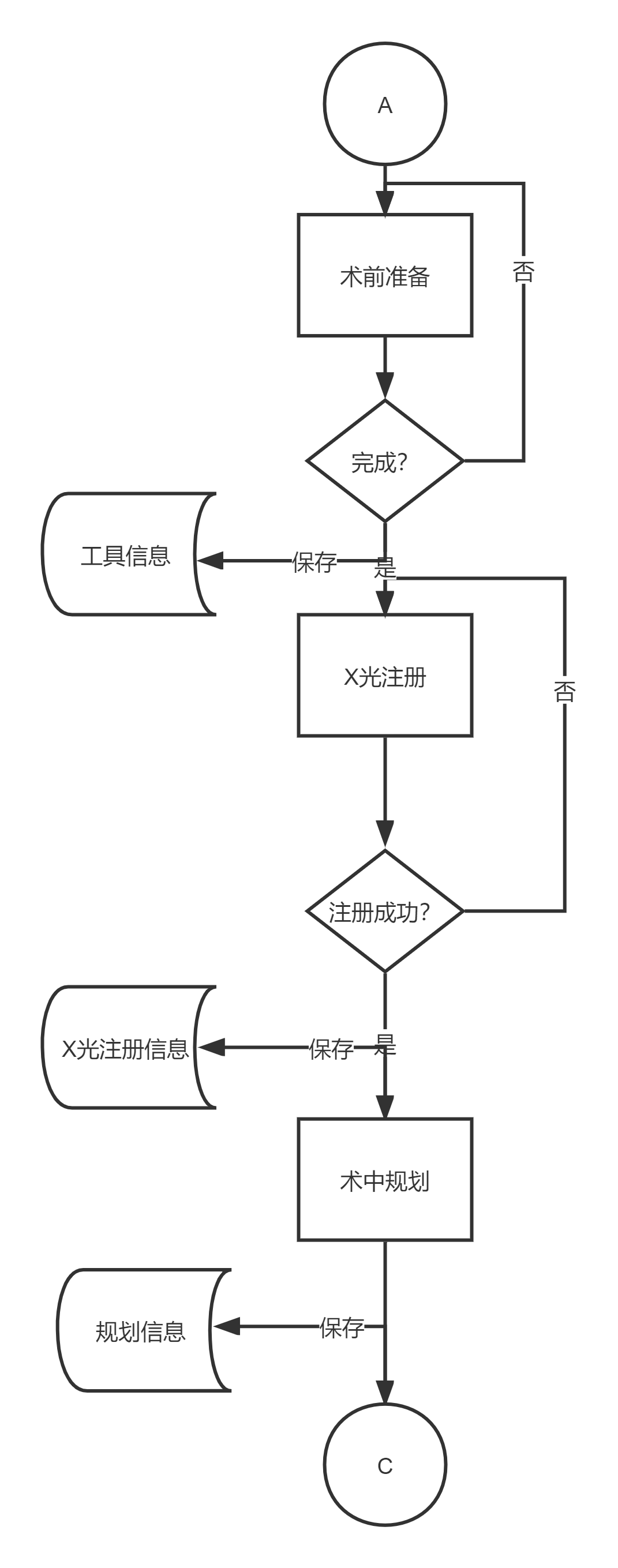
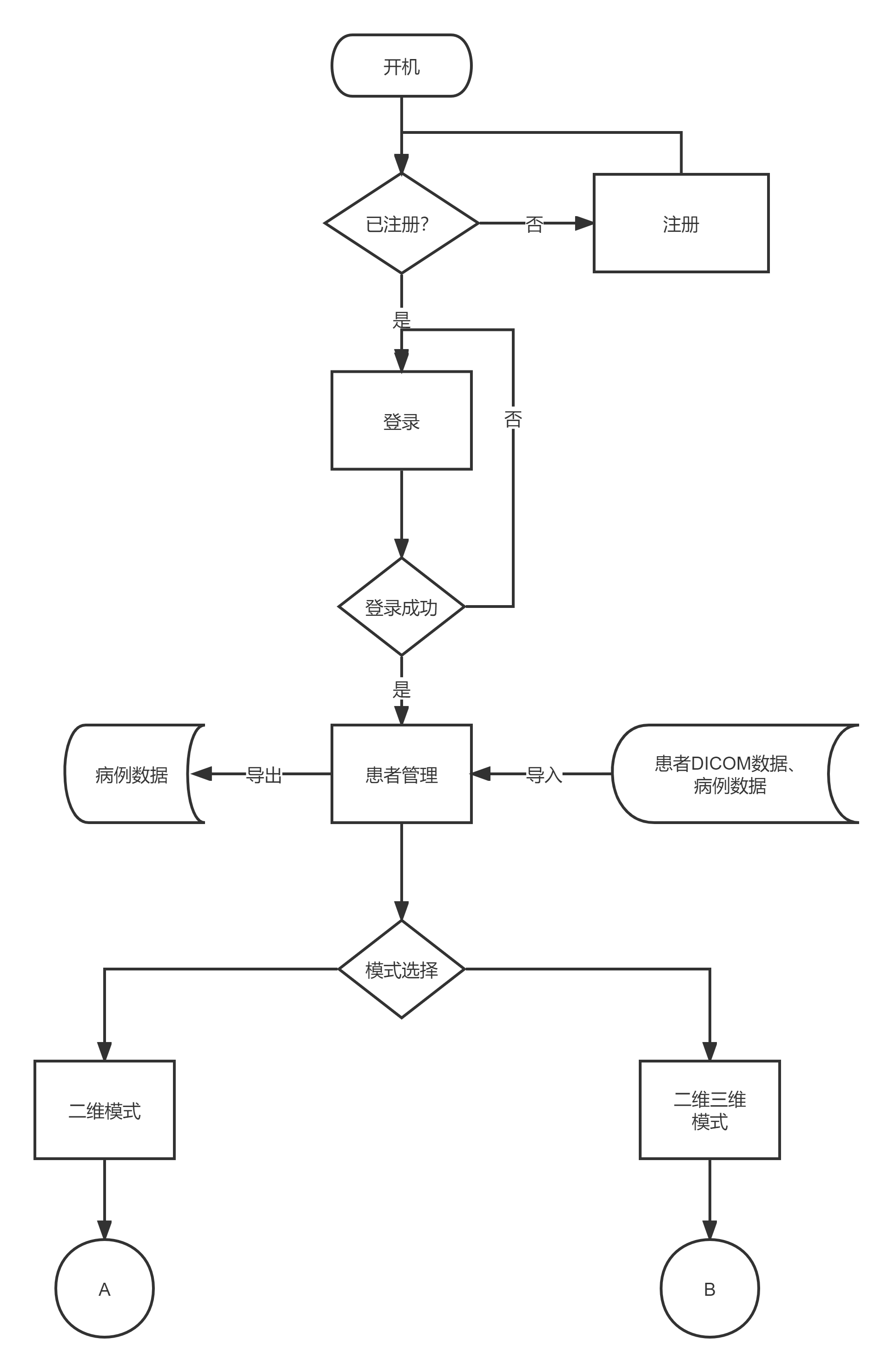
|  |  |
| --- | --- |
| 专有名词 | 描述 |
| 冠状位 | 为左,右方向将人体纵切为前后（腹背）两部分的断面。 |
| 矢状位 | 将躯体纵断为左右两部分的解剖平面。 |
| 横断位 | 将躯体纵断为上下两部分的解剖平面。 |
| 椎形投影 | 以点光源发射的椎形光束生成的投影图像。 |
| 平行投影 | 以平行光穿透组织产生的投影图像。 |
| DICOM | 一种医疗图像格式，包含患者信息，图片信息，成像设备信息等。 |
| DRR | 数字重建放射影像，利用椎形光束对CT影像进行投影得到二维图像。 |
| 影像重建 | 将术前CT的断层数据重建成三维体数据，并进行三维渲染显示。 |
| 手术规划 | 基于三维或二维医学图像计划手术植入物路径。 |
| 术中配准 | 匹配术前CT仿真透视图像（如DRR）与术中透视图，获取当前病灶（路径）的真实位姿。 |
| C臂机 | C形臂X片机。 |
| C臂机校准 | 通过术前大量采集透视图像（已知布局的棋盘格）来计算C臂内参及畸变参数，用于C臂图像的校正及透视变换。 |
| X片注册 | 对含有配准板的X透视图像，识别图像中钢珠球的二维坐标，并与钢珠球三维坐标进行配准，计算出配准板与Ｃ臂机间的空间变换关系。 |
| 仿真 | 根据机械臂的初始位置和目标位置，模拟机械臂的运动过程、并显示。 |
| 定位 | 根据3D目标位置，控制机械臂运动定位。 |
| 双目相机 | 双目红外光学相机。 |
| 机械臂 | 包含机械臂本体、控制箱和末端控制器。 |
| 示踪器 | 由四个红外反光球一组按特定结构设计排布构成能被双目相机跟踪位姿的硬件。 |

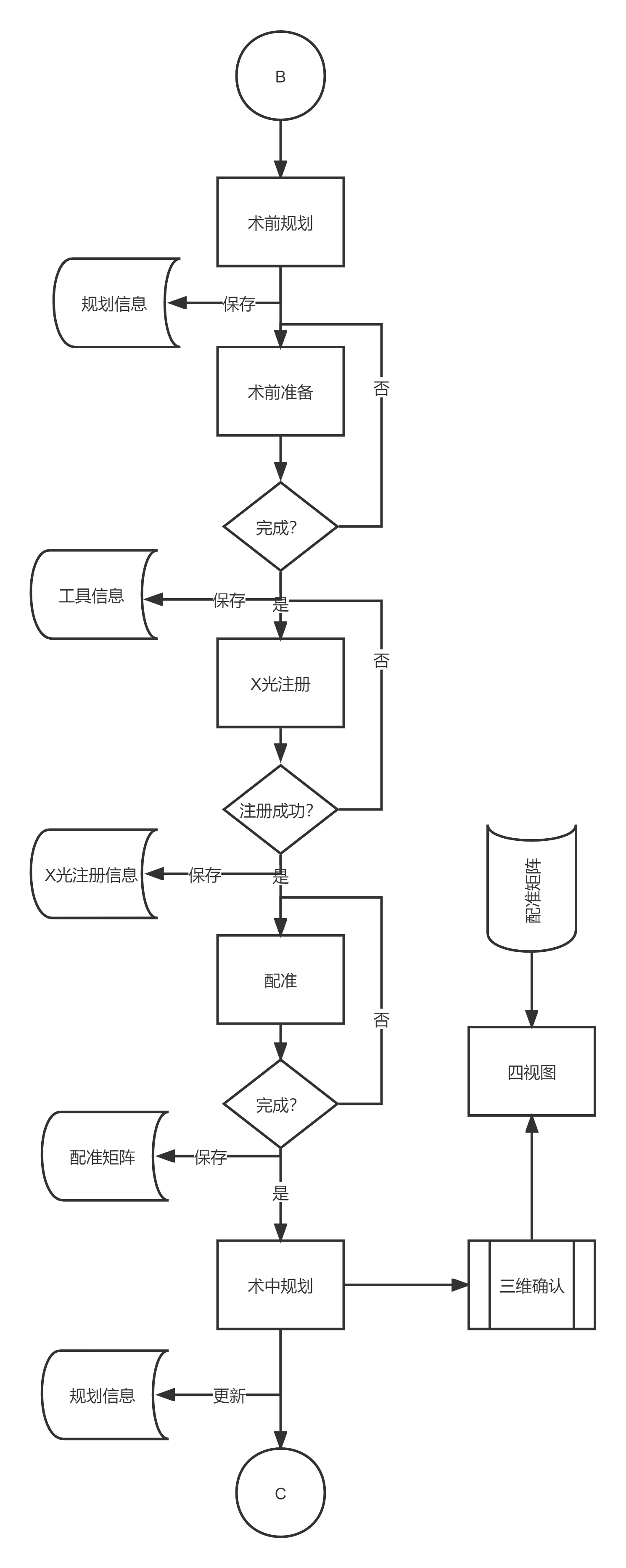
## 参考资料

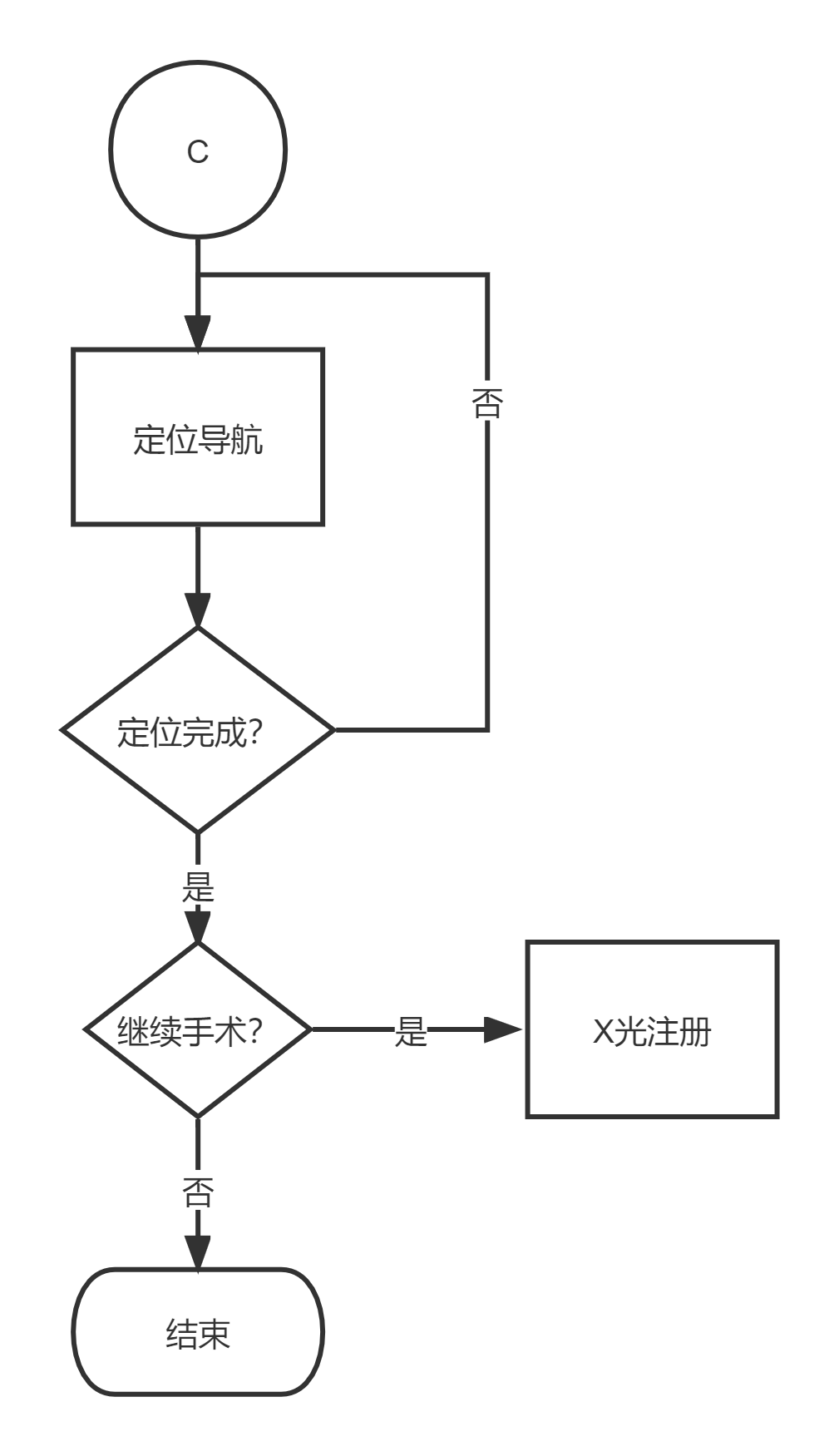
1. 《医疗器械软件注册技术审查指导原则》
2. 《医疗器械网络安全注册技术审查指导原则》
3. YY/T 0664-2020 《医疗器械软件生存周期过程》
4. YY/T 1406.1-2016 《医疗器械软件 第1部分：YY/T 0316 应用于医疗器械软件的指南》
5. GB/T 8567-2006《计算机软件文档编制规范》
6. 《MS-002P 技术需求规格书》
7. 《MS-002P 风险评估和控制记录》
8. 《MS-002P 软件概要设计说明书》

# 总体设计

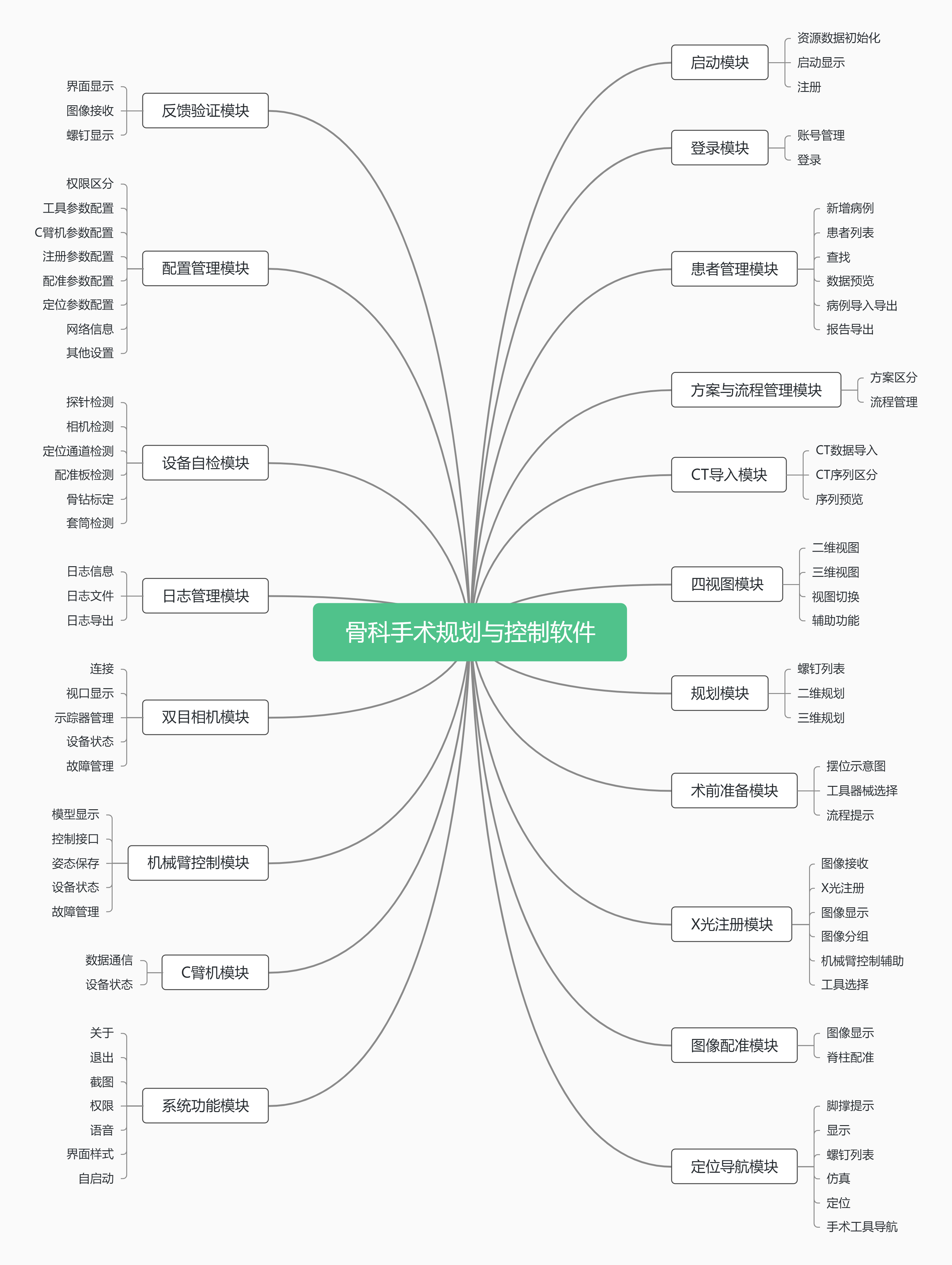
## 处理流程







## 总体结构和模块外部设计



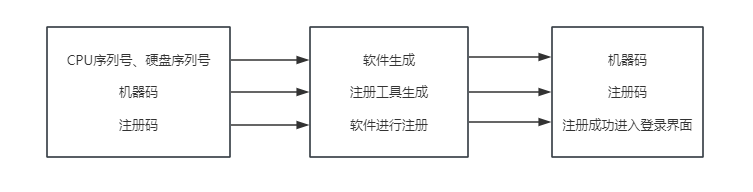
# 启动模块

## 模块描述

1. 软件启动是软件打开的第一步，保证了软件可运行；启动页面布局设计、风格合理；
2. 软件启动在软件中属于常驻内存、不可重入，不属于顺序处理且代码没有覆盖要求。本功能与其他功能耦合较小：启动成功成功可进入软件注册；
3. 为了防止软件被盗用，需要对软件进行保护；
4. 当系统软件在一台主机上安装后，会根据主机的硬件信息生成一串机器码
5. 需要一个专用工具。针对机器码及设定的密钥，生成注册码；
6. 将注册码录入，注册成功后方能使用，此时也将注册码写入到配置文件中。

## 功能

1. 软件启动时加载动画，执行播放；
2. 软件启动过程钟加载所需要的文件，启动NDI、机械臂等一系列外部连接；
3. 软件启动后自动跳转软件注册进行验证。
4. 验证码和机器码均为32位英文小写字母和阿拉伯数字的混合码
5. 进入软件后，读取配置文件中的注册码，验证是否正确；
6. 验证通过后，正常进入登录界面；
7. 首次安装等情况下，验证不成功，弹出注册对话框；其中机器码根据CPU系列号及硬盘序列号通过内部算法自动生成一串机器码；
8. 使用注册工具根据机器码，通过内部算法对应的注册码，将注册码输入注册界面进行注册，注册成功后将注册码写入配置文件中并进入登录界面。



## 性能

无。

## 输入项目

1. 硬件信息：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | CPU序列号 | cpuId | QByteArray | 16~32位 | 自动获取 | / |
| 2 | 硬盘序列号 | diskId | QByteArray | 8~32位 | 自动获取 | / |

1. 机器码：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | 机器码 | systemCode | QByteArray | 32位 | 手动输入 | / |

## 输出项目

1. 机器码：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出形式** | **备注** |
| 1 | 机器码 | systemCode | QByteArray | 32位 | 字符串 | / |

1. 注册码：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出形式** | **备注** |
| 1 | 注册码 | licensegenerated | QByteArray | 32位 | 字符串 | / |

## 算法

算法目的：

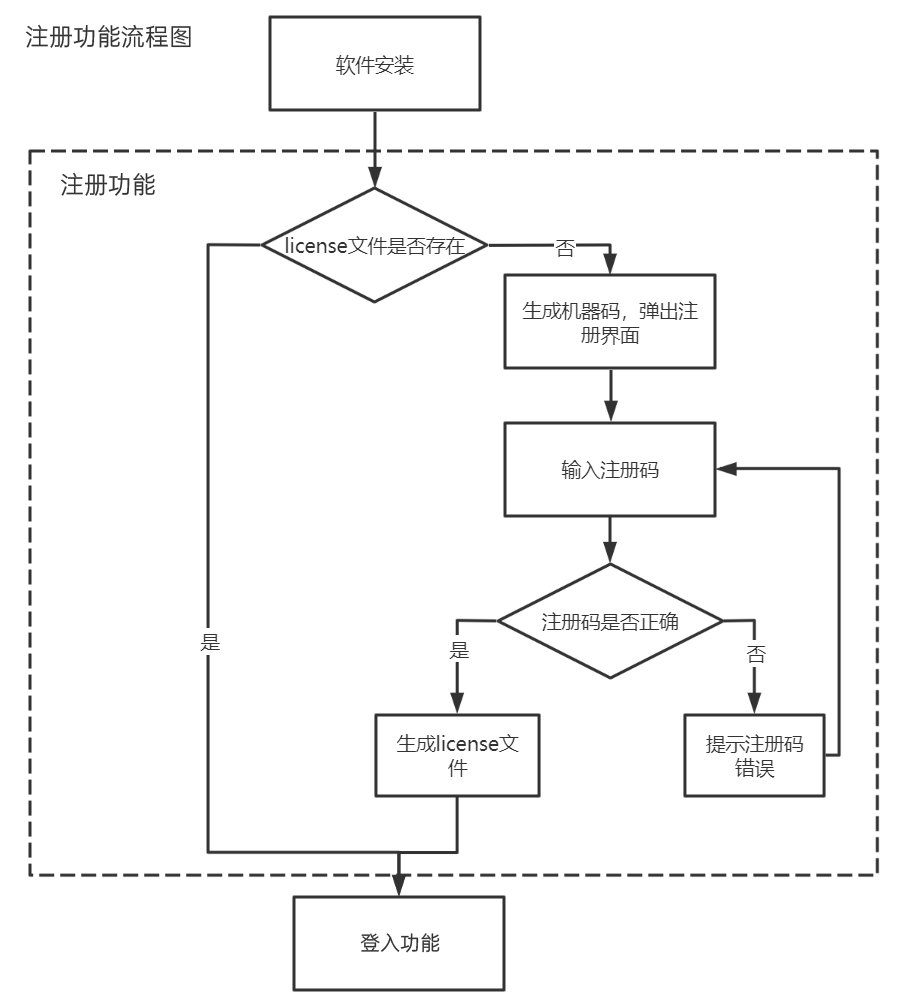
为了防止软件被盗用，需要对软件进行保护。

算法流程：

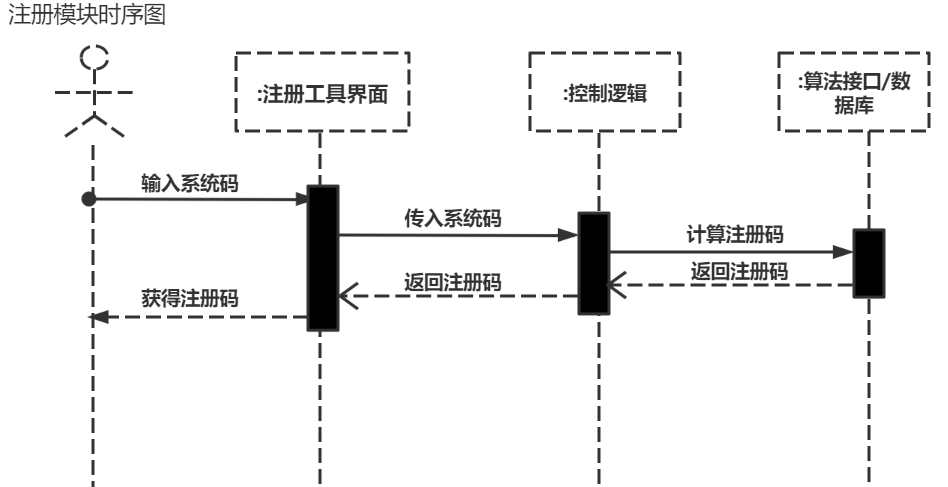
1. 软件启动后会检查配置文件license.json中是否有32位的注册码，如果不存或注册码无效则被判断为未注册。
2. 如果软件未注册，则程序会取出本机cpu序列号和硬盘序列号拼接，然后经过MD5函数生成MD5码即机器码提供给用户，同时生成随机密钥。
3. 在注册工具中输入机器码，注册工具会将机器码和密钥字符串经过Hash码生成32位的注册码提供给用户。
4. 将注册码输入到软件后，软件程序也会将机器码和密钥字符串经过Hash生成32位的注册码，然后与用户输入的注册码进行比对，如果相同，则注册成功。
5. 软件注册成功，程序同时也会将注册码写入到license.json，这样下次登入程序时，执行步骤 1）判断为已注册，程序直接进入登入界面。

## 程序逻辑

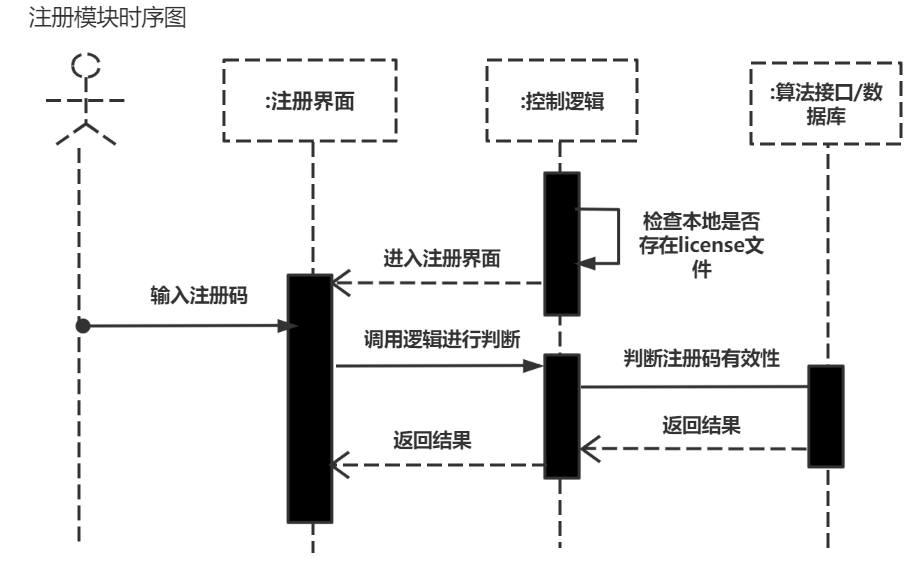
启动功能独立于其他功能，随软件启动而打开。



软件注册功能在软件内的流程图



软件注册功能获取注册码时序图



注册功能使用注册码注册时序图

## 接口

软件启动功能包含StartupWidget这个类，注册功能包含SystemInfo、SoftLicense、Register三个主要类。其中重要接口详细说明如下。

3.8.1. StartupWidget

/\*

\* 功能描述：初始化日志文件

\*/

void InitLogger();

/\*

\* 功能描述：初始化NDI相机

\*/

void InitNDI();

/\*

\*功能描述： 初始化机械臂

\*/

void InitRobot();

/\*

\* 功能描述：设置配置文件数据

\*/

void SettingData();

3.8.2. SystemInfo

/\*

\* 功能描述：获取硬盘序列号

\*/

void diskId();

/\*

\* 功能描述：获取CPU序列号

\*/

void cpuId();

3.8.3. SoftLicense

/\*

\* 功能描述：生成机器码测试

\*/

void systemCode();

/\*

\* 功能描述：判断注册码的有效性测试

\* 返回值：true,验证通过；false,验证不通过

\*/

bool isValid();

/\*

\* 是否生成注册码测试

\*/

void generateLicense();

3.8.4. Register

/\*

\* 功能描述：是否生成机器码

\*/

void registerLicense();

/\*

\* 功能描述：点击取消注册

\*/

void cancelClicked();

/\*

\* 功能描述：点击关机

\*/

void shutdownClicked();

## 存储分配

软件注册功能本地文件存储内容：保存在当前软件根目录下：“.// license.json”文件。

## 限制条件

Windows平台。

## 测试要点

1. 软件是否能正常打开。
2. 硬件正常，机器码可以生成；
3. 硬件去除，机器码生成失败；
4. 注册码正确，注册成功；
5. 注册码错误，注册失败；
6. 注册码格式正确，非本机生成，注册失败。

# 登录模块

## 模块描述

1. 登录功能是用户进入系统的唯一入口，保证了软件的安全；登录功能布局设计、风格合理；
2. 登录功能在软件中属于常驻内存、不可重入，不属于顺序处理且代码没有覆盖要求。本功能与其他功能耦合较小：登录成功可进入数据管理功能；
3. 用户信息以特定的内部加密方式保存在本地文件中，后台数据不允许普通用户进入和访问，防止数据泄露。
4. 不同账户拥有不同权限，保证软件网络安全的授权能力。
5. 用户拥有唯一账户，需要身份验证才能登陆，保证了软件网络安全的人员鉴别能力。

## 功能

1. 系统启动后自动开启软件，能进入登录界面。
2. 输入正确的账号密码后可以成功离开登录功能进入下一步。
3. 账号密码为英文和数字组合字符串，长度不小于6个字符且不大于10个字符。输入的密码长度大于10时字符无法输入。
4. 每次软件打开，账号栏默认选择操作员账号。
5. 每个账号对应一组密码。
6. 账号密码错误时，软件提示对应消息。
7. 软件用户通过不同账号、密码进行登录使用，根据不同账号，分配出不同权限对软件进行操作。
8. 模块有关机、锁屏、重启等相关功能，锁屏后自动调出锁屏界面，关机按钮关闭计算机，重启按钮可以重启电脑并打开软件。
9. 关于界面能打开软件相关信息，以及使用文档和维护手册等相关文档
10. 管理员账号和工程师账号进入系统后，在设置界面可添加用户和密码，新添加用户上限20个。

## 性能

无。

## 输入项目

登录功能-全功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 账号 | user\_id | varchar | 满足规定正则 | 下拉选择 | 无 |
| **2** | 密码 | user\_pwd | varchar | 满足规定正则 | 手动输入 | 无 |

## 输出项目

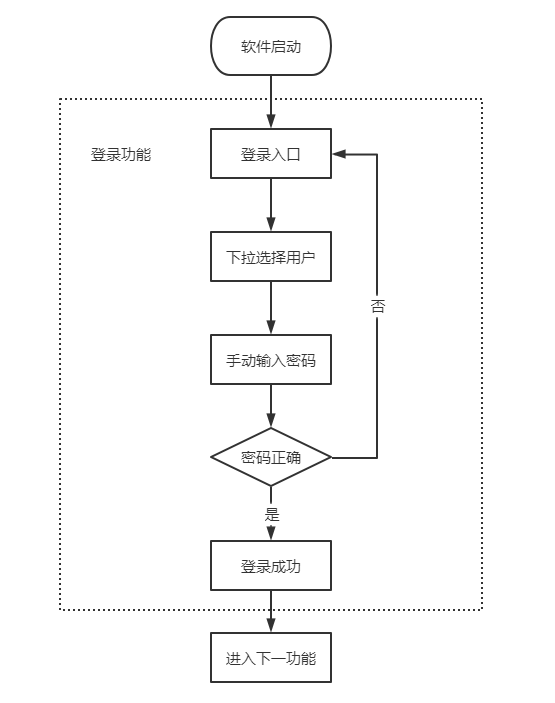
登录功能输出是从数据库中读取的数据或错误警告信息。

## 算法

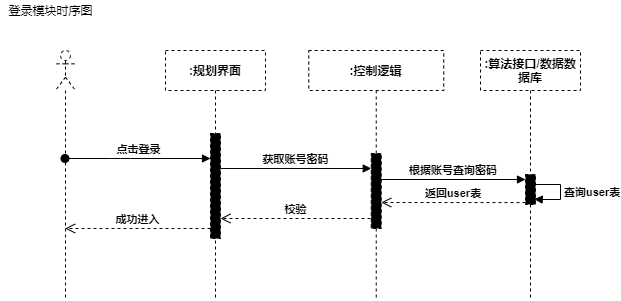
无。

## 程序逻辑

登录功能独立于其他功能，随软件启动而启动。



登录功能流程图



登录功能时序图

## 接口

登录功能主要包含LoginWidget。其中重要接口详细说明如下：

/\*

\* 功能描述: 登录功能

\*/

void LoginWidget::LoginButton\_clicked();

/\*

\* 功能描述: 关机锁屏等功能

\*/

void LoginWidget::PowerButton\_clicked();

/\*

\* 功能描述: 软件关于功能

\*/

void LoginWidget::AboutButton\_clicked();

/\*

\* 功能描述: 软件校验功能

\* 返回值：true,验证通过；false,验证不通过

\*/

bool LoginWidget::IsValidLicense();

## 存储分配

登录功能信息存在本地数据库中：软件所在配置目录下：”ms002db.db”文件。

## 限制条件

1. 密码格式必须是标准字符串且长度6-10位。
2. 密码为英文和数字组合字符串。

## 测试要点

1. 用户名密码正确，登录；
2. 密码为空，密码错误；
3. 密码小于6位，密码错误；
4. 密码6-10位有效字符验证，无效字符长度验证。

# 患者管理模块

## 模块描述

1. 患者管理模块是用户登陆系统后的首页，在这里对进行患者信息进行了展示，并可以新增、修改、导入和导出患者信息；
2. 在患者信息列表中展示的字段有：患者姓名、创建时间、病历号、配准模式；
3. 患者信息存储于数据库中，每个普通用户仅能看到当前用户的患者信息，工程师与管理员账号可以看到所有的患者信息，管理员账号仅能查阅患者信息，不能进行增加、修改、导入、导出等操作；
4. 在患者管理模块选择要进行规划及手术的患者，不选择患者则无法进行下一步。

## 功能

1. 登陆完成后即进入到患者管理界面，在患者管理界面会展示当前用户的患者信息（工程师账号、管理员账号则会展示所有的患者信息），包括患者姓名、创建时间、病历号、配准模式；
2. 列表中的患者数据信息将依据创建时间进行倒序排列；
3. 在上方搜索栏中可以通过患者姓名、创建时间、病历号、配准模式对患者信息进行搜索，在搜索框中输入搜索依据后点击搜索按钮后，患者列表中就会显示相匹配的患者信息；
4. 列表中的患者信息可以通过左键双击打开详细信息弹窗，在详细信息弹窗中可以对患者姓名、患者性别、病历号、患者年龄、备注进行修改和保存（管理员账号仅能查看，不能修改或保存）；
5. 可以通过单机选中患者，选中患者后会进行数据预览；
6. 对于配准模式为“纯二维配准模式”的患者，在已经加载了X片的情况下会在右侧展示其X片，预览窗口默认显示第一组X片图像，通过滑动条可以切换图像；
7. 对于匹配模式为“二维三维配准模式”的患者，在已经加载了CT的情况下，预览窗口会显示CT横断位图像，默认显示中间的CT，通过鼠标滚轮与滑条可以切换图层，中键移动鼠标可以对图像进行平移，通过滑动条可以调整窗位窗宽；
8. 点击新增病例按钮会弹出新增病例的弹窗，用户输入患者姓名、患者性别、配准模式等必填项后点击下一步即可生成患者编号并新增患者信息;
9. 若点击下一步时，有必填项信息未填写则给出相关提示；
10. 对于配准模式为二维三维配准模式的患者，点击下一步则进入CT导入模块；
11. 对于配准模式为纯二维配准模式的患者，点击下一步则进入术前准备模块；
12. 导出病例时，首先选中要导出的病例，然后点击导出病例，选择导出到的路径，点击确认后将患者相关的数据加密压缩后导出；
13. 导入病例时，点击导入病例后，选择要导入的病例数据，点击确认后，即可将病例数据解压导入；
14. 每个用户只能导入由当前用户或工程师账号创建的病例，工程师账号可以导入所有用户创建的病例，管理员账号不能导入病例；
15. 导入病例与添加患者之前会判断当前系统中的存储空间，磁盘空间大于10G才能继续进行操作
16. 对于已经完成的手术可以生成pdf格式的手术报告，导出软件。

## 性能

无

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | 患者数据 | files | 文件 | 符合软件内部数据格式 | 弹窗选择 | 选择患者数据 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | 患者姓名 | String | 字符串 | 12字节内可包含中英文、数字 | 输入框 | 必填 |
| 2 | 患者性别 | Enum | 枚举 | 男，女 | 单选框 | 必填 |
| 3 | 病历号 | String | 字符串 | 20字节内可包含英文、数字 | 输入框 | 选填 |
| 4 | 患者年龄 | Int | 整数 | [0,200] | 输入框 | 选填 |
| 5 | 备注 | String | 字符串 | 400字节以内 | 输入框 | 选填 |
| 6 | 配准模式 | Enum | 枚举 | 二维三维模式，二维模式 | 单选框 | 必填 |

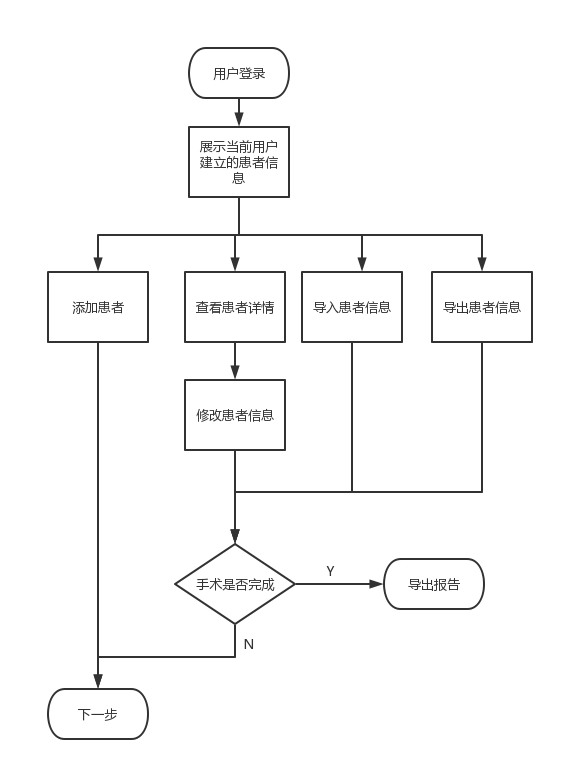
## 输出项目

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **标识** | **数据类型** | **有效范围** |
| 创建时间 | create\_time | Datetime | / |
| 医生姓名 | doctor\_name | String | / |
| 医院名称 | hospital\_name | String | / |
| 备注 | notes | String | 400字节以内 |
| 患者年龄 | patient\_age | Int | [0,200] |
| 患者数据 | patient\_datapath | String | / |
| 患者编号 | patient\_id | String | / |
| 患者姓名 | patient\_name | String | 12字节内可包含中英文、数字 |
| 患者性别 | patient\_sex | Enum | 男，女 |
| 病历号 | record\_num | String | 20字节内可包含英文、数字 |
| 配准模式 | surgery\_mode | Enum | 二维三维模式，二维模式 |

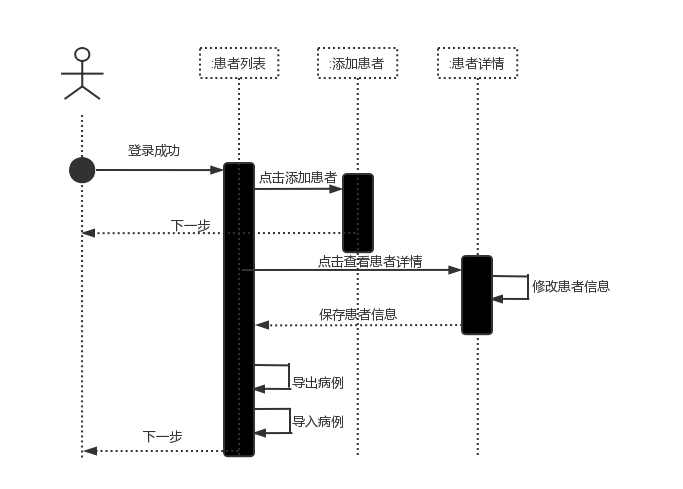
## 算法

无

## 程序逻辑



患者管理模块在软件内的流程图



患者管理模块时序图

## 接口

患者管理功能主要包含 PatientManagerWidget和PatientDetailInfoWidget。其中重要接口详细说明如下：

5.8.1. PatientManagerWidget

/\*

功能描述: 进度条提示窗口，可以设置进度，提示内容并进行确认或取消

\*/

class PatientManagerWidget :

/\*

\* 功能描述：初始化患者信息表格的表头，以及一些样式，并加载患者信息

\*/

void InitTable();

/\*

\* 功能描述：初始化信号槽

\*/

void InitSignalToSlot();

/\*

\* 功能描述：初始化界面

\*/

void InitWidget();

/\*

\* 功能描述：管理权限，对于不同的角色进行权限管理

\*/

void ManageAuthority();

/\*

\* 功能描述：更新患者信息数据列表

\*/

void UpdateTable();

5.8.2. PatientDetailInfoWidget

/\*

\* 功能描述: 患者详情页

\*/

class PatientDetailInfoWidget

/\*

\* 功能描述：根据id获取并展示患者信息,用于查看患者详情，新建患者则不必调用

\* 参数：id

\*/

void SetPatient(QString id);

/\*

\* 功能描述：初始化界面

\*/

void InitWidget();

/\*

\* 功能描述：新增用户

\* 返回值：true：新增成功，false：新增失败

\*/

bool AddPatient();

/\*

\* 功能描述：编辑用户

\* 返回值：true：修改成功，false：修改失败

\*/

bool EditPatient();

/\*

\* 功能描述：获取当前医院名称

\* 返回值：医院名称

\*/

QString GetHospital();

/\*

\* 功能描述：获取当前登录的医生

\* 返回值：当前登录的医生，如果为操作员、管理员、工程师则返回值为空

\*/

QString GetDoctor();

/\*

\* 功能描述：关联信号和槽函数

\*/

void ConnectSlotAndSignal();

/\*

\* 功能描述：表单校验，对表单中各项输入进行校验

\* 返回值：true,校验通过；false,校验未通过

\*/

bool Check();

/\*

\* 功能描述：检查字符串中的中文标点

\* 参数：被检查字符串

\* 返回值：true,含中文标点；false,不含中文标点

\*/

bool CheckChinesePunctuation(const QString& str);

/\*

\* 功能描述：将表单中的患者信息同步到实体类中

\*/

void UpdatePatient();

## 存储分配

1. 患者数据列表保存在当前程序目录：”.\config\patient\_table\”文件夹下；
2. 每个患者的CT数据存储在当前患者目录：”.\data\患者id\CT\”文件夹下；
3. 每个患者的X片数据存储在当前患者目录：”.\data\患者id\Xray\”文件夹下；
4. 进入到下一步的患者数据存储在： FMODEL(MainModel)->patient\_。

## 限制条件

1. 患者姓名由12字节以内的汉字、字母及数字组成；
2. 患者年龄在[0,200]之间；
3. 患者病历号由20字节以内的字母、数字组成；
4. 患者备注在400字节以内；
5. 新增患者与导入患者时要求系统内控件大于10G;
6. 新增患者与修改患者信息时患者姓名、患者性别、配准模式为必填项。

## 测试要点

1. 患者数据列表是否正常显示；
2. 患者增加测试；
3. 患者信息修改测试；
4. X片、CT预览是否正常显示；
5. 患者导入测试；
6. 患者导出测试；
7. 磁盘空间不足时，添加患者与导入患者测试；
8. 未连接U盘时，导入病例与导出病例测试；
9. 按条件搜索患者信息测试；

# 方案与流程管理模块

## 模块描述

1. 方案与流程管理模块主要用于管理手术流程，根据不同的手术模式，选择对应的手术流程。

## 功能

1. 根据病例的配准方式确定方案，根据方案确定对应的手术流程；
2. 流程步骤只允许线性跳转，通过上一步下一步按钮跳转；
3. 方案流程窗口具有导航栏，能够显示当前以及前后步骤。

## 性能

无

## 输入项目

无

## 输出项目

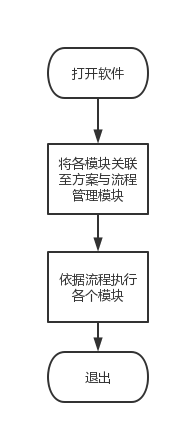
无

## 算法

无

## 程序逻辑

方案与流程管理模块，主要用于管理其他模块的使用顺序。



方案与流程管理模块在软件内的流程图

## 接口

方案与流程管理模块包含StepManager这个主要类。其中重要接口详细说明如下：

/\*

\* 功能描述:软件步骤流程管理

\*/

class StepManager

/\*

\*功能描述: 单例

\*返 回 值: 单例对象

\*/

static StepManager\* instance();

/\*

\*功能描述: 获取当前步骤

\*返 回 值: PrimaryStep：步骤

\*/

PrimaryStep GetCurrentStep();

/\*

\*功能描述: 设置当前步骤

\*参 数:

\*\*参数0: step：步骤

\*/

void SetCurrentStep(PrimaryStep step);

/\*

\*功能描述: 获取当前手术模式

\*返 回 值: OperationScheme：手术模式

\*/

OperationScheme GetCurrentScheme();

/\*

\*功能描述: 设置当前手术模式

\*参 数:

\*\*参数0: scheme:手术模式

\*/

void SetCurrentScheme(OperationScheme scheme);

/\*

\*功能描述: 下一步

\*/

void NextStep();

/\*

\*功能描述: 上一步

\*/

void LastStep();

## 存储分配

无

## 限制条件

无

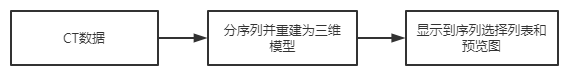
## 测试要点

1. 手术流程是否正常；
2. 导航栏内容是否合理；
3. 点击上一步是否能够正常返回，点击下一步是否能进到下一步。

# CT导入模块

## 模块描述

CT导入模块主要用于从U盘选择并导入CT数据，并将CT分序列、重建为三维数据加载到内存，为后续模块提供输入。CT导入模块包含CT导入界面及相关逻辑。CT导入界面包含“导入CT”、“返回”、“下一步”按钮，有图像序列选择窗口和图像预览窗口。模块IPO图如下:



CT导入模块IPO图

## 功能

1. 点击“导入CT”按钮可弹出文件夹选择框，可在文件夹选择框选择数据所在文件夹。 如果选择的文件夹及其下一级子文件夹都无DICOM数据，会提示“无CT数据，请重新加载”。
2. 加载文件夹目录和下一级子目录中所有DICOM文件，支持最大导入10GB的数据，大于10GB不导入，并提示“数据大于10GB，无法完全导入”。
3. 加载的过程中有进度条提示。
4. 通过DICOM标签SeriesInstanceUID区分序列。
5. 加载数据，如果数据有断层，提示“CT序列中有错误，请更换CT”，界面还原到加载数据前。如果导入的CT路径长度大于255个字符，提示“导入CT路径过长”，界面还原到加载数据前。
6. 序列选择窗口，包含姓名、出生日期、扫描时间、CT序列号、CT层厚、本序列层数和CT缩略图，每个序列的缩略图为序列中间的那张。
7. 序列选择窗口中默认选中第一个CT序列，选中序列后，序列框高亮显示。
8. 预览窗口显示选中序列的二维图像，鼠标滚轮可切换图层，点住中键移动鼠标对图像进行平移，点住右键移动鼠标对图像进行放大缩小，有调整窗宽窗位的滑条，窗口右侧有滑动条，鼠标左键拖动滑动条切换图层。
9. 选中序列后，点击“下一步”按钮，如果所选序列单序列层数小于5，提示“CT层数过少，不允许载入，请选择其它序列”。如果所选序列数据层厚不满足小于1mm等条件，提示“CT层厚大于1mm，不允许载入，请选择其它CT”。如果选中序列没有任何问题，点击“下一步”按钮会进入X片注册界面，选中的CT序列保存至患者的数据文件夹中。
10. 点击“返回”按钮，返回患者管理界面，CT数据不保存，下次进入该界面重新选择CT数据。

## 性能

1. 三维数据重建速度≥15MPS。

## 输入项目

CT导入功能-全功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | ct序列 | files | 文件 | 符合DICOM3.0格式 | 自动输入 | U盘导入 |

## 输出项目

CT导入功能-全功能输出项目如下：

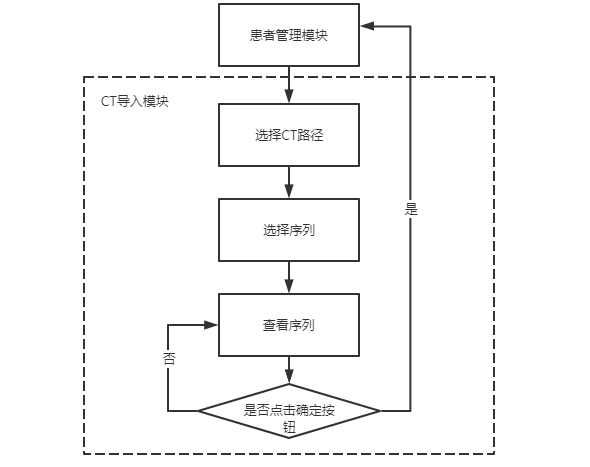
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | 重建出来的vtk数据 | vtkSmartPointer<vtkImageData> | vtkImageData  的指针对象 | 不为空 | 自动保存 | 被选中序列重建数据 |
| **2** | 被选中序列对应的序列文件 | files | dcm | 文件数不为0 | 自动保存 | 序列文件 |

## 算法

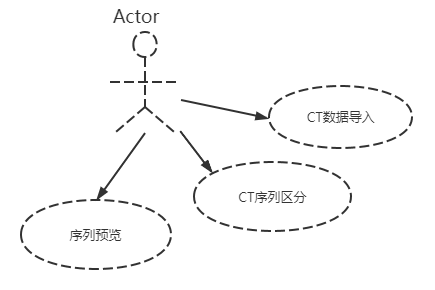
CT导入不包含算法。

## 程序逻辑

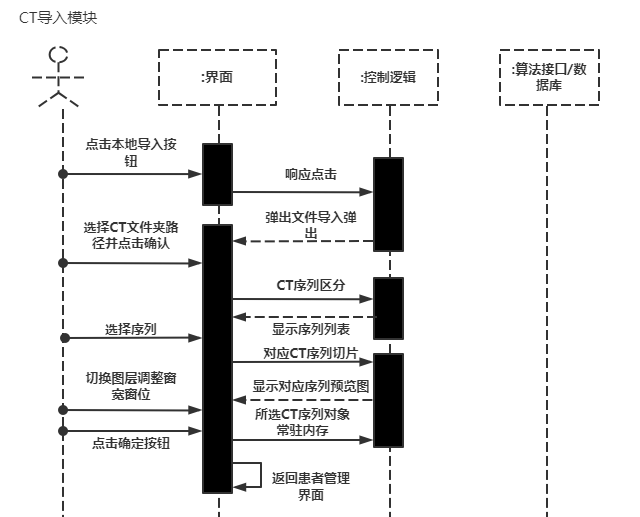
CT导入是本软件顺序流程的一个步骤，上一步为患者管理，下一步为X-Ray注册。分别用CT导入功能在软件内的流程图、CT导入功能系统用例图、CT导入功能时序图来展示本功能的程序逻辑。



CT导入模块在软件内的流程图



CT导入模块用例图



CT导入模块时序图

## 接口

CT导入包含多个主要类。其中重要接口详细说明如下：

/\*

\*功能描述: CT导入

\*参 数:

\*\*参数0: CT路径

\*返 回 值: 无

\*/

void ImportCtByPath(QString &ctPath);

/\*

\*功能描述: 载入CT序列

\*参 数:

\*\*参数0: 序列索引

\*返 回 值: 无

\*/

void LoadDicomSeriesData(int seriesIndex);

/\*

\*功能描述: 渲染窗口响应交互

\*参 数:

\*\*参数0: vtk实体对象指针

\*\*参数1: 事件

\*\*参数2: 数据指针

\*\*参数3: 回调数据指针

\*返 回 值: 无

\*/

void EventCallBack(vtkObject\* object, unsigned long event, void\* clientdata, void\* calldata)

/\*

\*功能描述: 渲染窗口刷新

\*参 数:无

\*返 回 值: 无

\*/

void Render()

/\*

\*功能描述: 响应导入按钮

\*参 数:无

\*返 回 值: 无

\*/

void on\_choice\_ct\_btn\_clicked()

## 存储分配

CT导入本地文件存储内容：

1. 选择的CT序列对应的DICOM文件保存在当前患者目录下： "./CT"目录下。

规划功能内存存储内容：

1. 序列重建数据（全局唯一）: FMODEL(Orth2DModel)->ct\_image；

## 限制条件

1. 所选CT文件夹存在；
2. CT文件夹内有数据
3. CT文件夹内的数据可以被DICOM3.0协议读取；

## 测试要点

1. 读取中文的CT文件夹路径；
2. 2D视图鼠标交互测试，窗宽窗位重置；
3. 序列区分是否正确；

# 四视图模块

## 模块描述

四视图模块主要用于显示CT不同切面的二维图像和CT重建后的三维图像，作为规划模块的基础。四视图模块包括二维视图、三维视图、视图切换功能。

## 功能

1. 包含四个窗口，分为2+2与3+1两种排列模式，通过窗口内按钮切换显示。
2. 二维视图对应显示CT数据横断位、矢状位和冠状位切面图像。
3. 二维视图内鼠标滚轮可切换图层，点住滚轮移动鼠标对图像进行平移，点住右键滑动可对图像放大缩小，相关按钮可以调整图像窗宽窗位以及恢复视图显示状态。
4. 左键点击其中一个二维视图中的任意位置，另外两个二维视图跳转显示到对应图层。
5. 三维视图对CT的三维模型进行渲染，有渲染模式切换功能，包括透视模式、骨骼模式、软组织模式与DRR模式，默认显示骨骼模式。
6. 三维渲染窗口中鼠标左键滑动为旋转操作，滚轮调整大小，点住滚轮移动鼠标对图像进行平移，鼠标右键左右滑动调整窗宽，上下滑动调整窗位。
7. 视图视角调整按钮，对渲染视角进行不同轴向的90°调整，预设10中颜色可以通过选择按钮对透视模式的渲染颜色进行更改，重置按钮恢复视图视角与渲染为默认状态。
8. 可在二维视图上进行长度测量和角度测量。长度测量，可画出刻度线，标出长度值。角度测量，画出角度刻度线，标出角度值。只能画一个长度和一个角度，长度和角度可同时存在。可清除长度和角度。

## 性能

1. 对性能无特别要求。

## 输入项目

四视图-全功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | CT重构结果 | 无 | vtkImageData | 不为空 | 自动输入 | 病例载入功能重构结果 |

## 输出项目

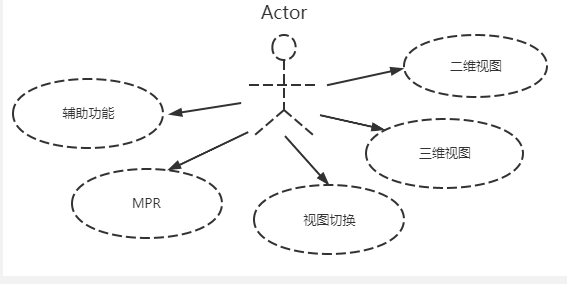
四视图主要功能是显示CT三维视图和二维切面，没有输出项目

## 算法

四视图模块不包含算法。

## 程序逻辑

四视图模块是规划模块的基础，提供二维视图、三维视图、视图切换功能、MPR功能和辅助功能，用用例图表示。



四视图模块用例图

## 接口

四视图模块包含多个主要类。其中重要接口详细说明如下：

/\*

\*功能描述: 切换布局模式

\*参 数:

\*\*参数0: 布局模式

\*返 回 值: 无

\*/

void SwitchLayoutMode(LAYOUT\_MODE mode);

/\*

\*功能描述: 初始化渲染窗口

\*参 数:

\*\*参数0: 当前视图索引

\*\*参数1: 窗口标签

\*返 回 值: 无

\*/

void InitRenderWindow(int thisViewIndex, QString label);

/\*

\*功能描述: 渲染窗口响应交互

\*参 数:

\*\*参数0: vtk实体对象指针

\*\*参数1: 事件

\*\*参数2: 数据指针

\*\*参数3: 回调数据指针

\*返 回 值: 无

\*/

void EventCallBack(vtkObject\* object, unsigned long event, void\* clientdata, void\* calldata)

/\*

\*功能描述: 渲染窗口刷新

\*参 数:无

\*返 回 值: 无

\*/

void Render()

/\*

\*功能描述: 响应测量长度按钮

\*参 数:无

\*返 回 值: 无

\*/

void on\_measure\_length\_btn\_clicked()

/\*

\*功能描述: 响应测量角度按钮

\*参 数:无

\*返 回 值: 无

\*/

void on\_measure\_angle\_btn\_clicked()

## 存储分配

四视图模块无文件存储内容。

## 限制条件

1. 长度测量只能在2D视图上进行；
2. 角度测量只能在2D视图上进行；
3. 四视图无法无限制缩放；
4. FMODEL(Orth2DModel)->ct\_image不能为空

## 测试要点

1. 查看2D视图，3D视图是否正常显示；
2. 2D视图鼠标交互测试，窗宽窗位重置；
3. 3D视图鼠标交互测试，窗宽窗位重置；
4. 3D颜色修改；
5. 3D渲染方式修改；

# 规划模块

## 模块描述

1. 规划功能是本软件实现术前规划功能的重要途经，是整个软件的核心功能。用于术前和术中路径的规划，在X-Ray或CT切片上自定义手术中放置螺钉，调整螺钉的位置、直径、长度等；
2. 规划功能在软件中属于常驻内存、不可重入，属于顺序处理且代码没有覆盖要求。本功能与其他功能耦合较小：利用全局静态指针获取病例功能X-Ray注册的数据，自身实现螺钉的规划。和定位功能复用同一个螺钉数据模型实现规划功能和定位功能螺钉数据的同步；
3. 功能内部代码增加边缘性检测，详见接口部分。

## 功能

二维规划

1. 二维规划界面有三个X片显示窗口、螺钉列表、二维规划功能按钮和“四视图”按钮。
2. 显示三个X片二维图像窗口，X片图像位置与视口相机位置与X片注册参数保持一致，前两个视口允许交互规划，第三个作为验证。
3. 窗口通过鼠标和功能按钮可以调整图像显示效果。三个窗口里有X片图像处理栏，默认收起以“图像处理栏”按钮显示，点击“图像处理栏”按钮图像处理栏展开，展开里面是“图像旋转”按钮、“图像还原”按钮、“图像对比度调解”按钮、“图像去小球”按钮、“图片增强”按钮、“图片反色”按钮和“图像处理栏收缩”按钮。这些图像处理按钮对应的图像处理功能全部可作用于当前显示窗口里的X片，点击“图像还原”按钮，X片可被还原。
4. 通过添加按钮触发以及鼠标交互添加螺钉到视图与螺钉列表，并通过二维视图鼠标交互以及功能按钮调整螺钉位置，以X片图像的空间位置以及螺钉在X片图像的投影图像判断螺钉的合理性。
5. 二维视图规划显示包含调整辅助线、规划轴线以及螺钉模型。螺钉有三个控制点，第一个控制点与螺钉头端（尖端）重合；第二个控制点与螺钉的尾端不重合，但始终在同一直线上；第三个控制点为前两个点的中点，通过三个控制点改变螺钉的位置和方向，不改变螺钉的长度。在图像中改变三个控制点，实际是在三维空间中操作控制点改变螺钉的位置和方向。
6. 点击“添加螺钉”按钮，在前两个视图中的一个视图中画出一根线，在另外两个视图中画出这根线对应的投影辅助线；然后在前两个视图中的另一个视图的辅助线内再画一根线，根据画的两根线计算出螺钉的三维坐标，在另外两张图像中画出该条线的投影辅助线，并根据三维坐标将螺钉的投影图像绘制在三张图像中，要能清楚的区分头和尾巴；第三张图像中无法调整螺钉的位置。螺钉列表中增加一个数据，并选中该数据，高亮显示。
7. 螺钉长度为当前生成长度最接近，且不大于生成长度的数值。例如生成长度为46.6mm，螺钉长度应该为45mm。如果生成长度小于5mm，螺钉长度为5mm。
8. 对已添加的螺钉修改长度时，螺钉头端位置不变，尾端在两个控制点所在直线上变化。
9. 螺钉列表包含名称、长度、直径、颜色、显示状态、样式切换、锁定状态、删除。锁定状态可在“锁定和解锁”间切换。锁定时，该通道不可调整，辅助线不可见。可一键全部锁定，选中当前螺钉默认解锁状态，未选中的全是锁定状态。
10. 点击螺钉列表中的“删除”按钮，弹窗提示“是否删除该螺钉”，点击“确认”后删除对应螺钉。
11. 添加螺钉以后，螺钉名称自动生成，默认为螺钉样式、绿色。
12. 螺钉的微调按钮。“上”、“下”、“左”、“右”四个按钮控制线的三个控制点。
13. 选中螺钉，在键盘上按住“Ctrl+V”可复制螺钉，即复制生成的螺钉与原螺钉的位置、长度、直径、颜色、显示状态和锁定状态相同。

三维规划

1. 三维规划界面包含四视图窗口、螺钉列表、“添加螺钉”、“复制螺钉”、“删除螺钉”按钮、“长度测量”按钮、“角度测量”按钮、“清除”按钮。
2. 四视图窗口包含所有8.2四视图模块功能。
3. 点击“长度测量”按钮和“角度测量”按钮可触发四试图模块里的测量功能，详见8.2四视图模块功能。点击“清除”按钮清除四视图中二维窗口中的测量线。
4. 通过“添加”按钮触发以及鼠标交互添加螺钉到视图与螺钉列表，并通过鼠标交互调整螺钉位置，以三维断层数据与二维断层数据判断螺钉的合理性。
5. 螺钉有三个控制点，第一个控制点为添加螺钉时点击的点，与螺钉头端（尖端）重合；第二个控制点为释放鼠标时的点，该点与螺钉的尾端重合；第三个控制点为螺钉的中点。通过两端的控制点改变螺钉的位置和方向，也改变螺钉的长度。选中的螺钉显示三个控制点和螺钉断层，未选中的螺钉只显示螺钉断层。
6. 螺钉长度为当前生成长度最接近，且不大于生成长度的数值。例如生成长度为46.6mm，螺钉长度应该为45mm。如果生成长度小于5mm，螺钉长度为5mm。
7. 对已添加的螺钉修改长度时，螺钉头端位置不变，尾端在两个控制点所在直线上变化。
8. 螺钉列表包含名称、长度、直径、颜色、显示状态、样式切换、锁定状态。锁定状态可在“锁定和解锁”间切换。锁定时，该通道不可调整，螺钉控制点不可见。可一键全部锁定，选中当前螺钉默认解锁状态，未选中的全是锁定状态。
9. 点击螺钉“删除”按钮，弹窗提示“是否删除该螺钉”，点击“确认”后删除对应螺钉。
10. 添加螺钉以后，螺钉名称自动生成，默认为螺钉样式、绿色。
11. 选中螺钉，在键盘上按住“Ctrl+V”或点击界面上的“复制螺钉”按钮可复制螺钉，即复制生成的螺钉与原螺钉的位置、长度、直径、颜色、显示状态和锁定状态相同。
12. 三维规划界面作为术前规划，作为主流程的一个界面。同时，在二维规划界面，点击“四视图”按钮会弹出三维规划弹窗。

## 性能

1. 规划螺钉数量≤50。

规划功能-全功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | X-Ray | image\_data\_ | vtkImageData | 不为空 | 外部传入 | 常驻内存 |
| **2** | CT | image\_data\_ | vtkImageData | 不为空 | 外部传入 | 常驻内存 |
| **3** | 通道头部位置 | pos1 | double[3] | 不为空 | 自动输入 | 常驻内存 |
| **4** | 通道尾部位置 | pos2 | double[3] | 不为空 | 自动输入 | 常驻内存 |
| **5** | 规划通道长度 | length | double | 大于0 | 自动输入 | 常驻内存 |
| **6** | 规划通道直径 | diameter | double | 大于0 | 自动输入 | 常驻内存 |
| **7** | 规划通道颜色 | color | color[3] | RGB值 | 自动输入 | 常驻内存 |
| **8** | 规划通道名称 | name | QString | 不为空 | 自动输入 | 常驻内存 |
| **9** | 是否显示为螺钉样式 | screw\_nut\_show | bool | true/false | 自动输入 | 常驻内存 |
| **10** | 规划通道是否显示 | channel\_show | bool | true/false | 自动输入 | 常驻内存 |

## 输出项目

规划功能-全功能输出项目如下：

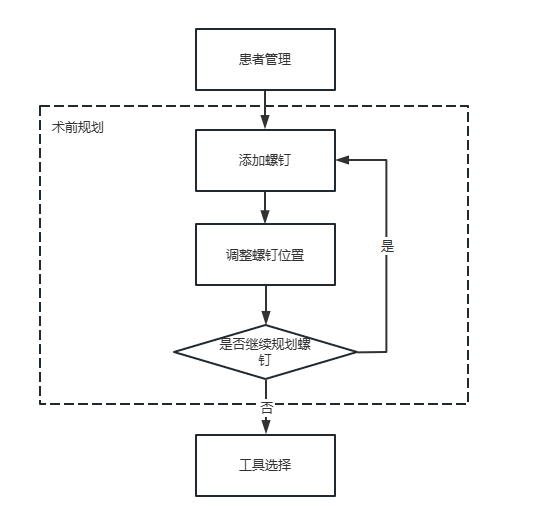
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | 规划通道头部位置 | pos1 | double[3] | 不为空 | 内存变量 | 常驻内存 |
| **2** | 规划通道尾部置 | pos2 | double[3] | 不为空 | 内存变量 | 常驻内存 |
| **3** | 规划通道长度 | length | double | 大于0 | 内存变量 | 常驻内存 |
| **4** | 规划通道直径 | diameter | double | 大于0 | 内存变量 | 常驻内存 |
| **5** | 规划通道颜色 | color | color[3] | RGB值 | 内存变量 | 常驻内存 |
| **6** | 规划通道名称 | name | QString | 不为空 | 内存变量 | 常驻内存 |
| **7** | 是否显示为螺钉样式 | screw\_nut\_show | bool | true/false | 内存变量 | 常驻内存 |
| **8** | 规划通道是否显示 | channel\_show | bool | true/false | 内存变量 | 常驻内存 |

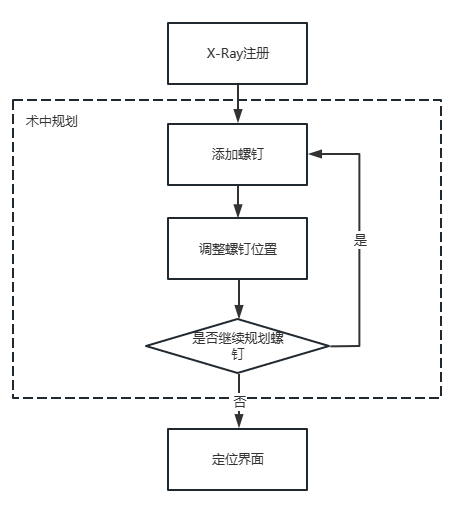
## 算法

规划功能不包含算法。

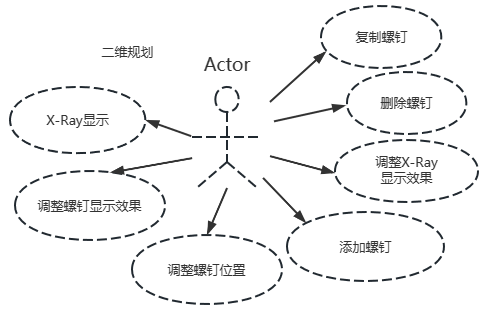
## 程序逻辑

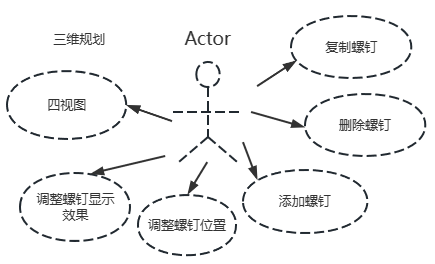
规划功能是本软件顺序流程的步骤,分为术前规划和术中规划。术前规划上一步为患者管理，下一步为工具选择。术中规划上一步为X-Ray注册，下一步为定位。分别用规划功能在软件内的流程图、规划功能系统用例图、规划功能时序图来展示本功能的程序逻辑。



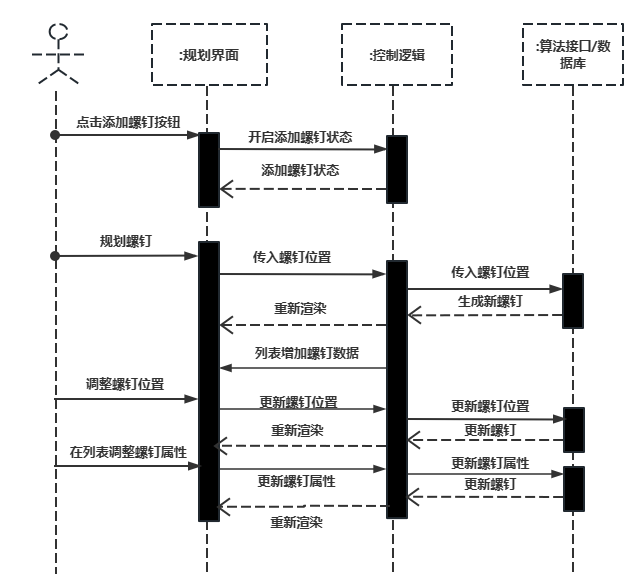


规划功能在软件内的流程图





规划功能系统用例图



规划功能时序图

## 接口

规划模块包含多个主要类。其中重要接口详细说明如下：

/\*

\*功能描述: 移动螺钉

\*参 数:

\*\*参数0: 方向

\*返 回 值: 无

\*/

void MoveScrew(int direction);

/\*

\*功能描述: 保存规划数据到json

\*参 数:无

\*返 回 值: 无

\*/

void WriteMyPlan2Json();

/\*

\*功能描述: 恢复通道数据

\*参 数:

\*\*参数0: 通道对象

\*返 回 值: 无

\*/

void RecoverPassageWay(passage\_ways::PassageWays\* passage\_way);

/\*

\*功能描述: 渲染窗口刷新

\*参 数:无

\*返 回 值: 无

\*/

void Render()

/\*

\*功能描述: 添加通道

\*参 数:无

\*返 回 值: 无

\*/

void AddPassageWay()

/\*

\*功能描述: 改变通道颜色

\*参 数:

\*\*参数0: 通道对象

\*\*参数1: 改变颜色

\*返 回 值: 无

\*/

void ChangePassageWayColor(passage\_ways::PassageWays\* passage\_way);

/\*

\*功能描述: 改变通道直接

\*参 数:

\*\*参数0: 通道id

\*\*参数1: 通道直径

\*返 回 值: 无

\*/

void ChangePassageWayDiameter(int screw\_id, double screw\_diameter);

/\*

\*功能描述: 改变通道可见性

\*参 数:

\*\*参数0: 通道id

\*\*参数1: 通道是否可见

\*返 回 值: 无

\*/

void ChangePassageWayVisible(int screw\_id, bool b\_screw\_channel\_show);

/\*

\*功能描述: 锁定通道

\*参 数:

\*\*参数0: 通道id

\*\*参数1: 通道是否锁定

\*返 回 值: 无

\*/

void ChangePassageWayLocker(int screw\_id, bool b\_screw\_locker\_show);

## 存储分配

规划功能本地文件存储内容：

已规划的数据保存在当前病例目录下： "./operation\_data\_json.json"文件。

规划功能内存存储内容：

1. 已规划螺钉数据（全局唯一）: FMODEL(Orth2DModel)；
2. 螺钉模型数据: FMODEL(ScrewModel)；

## 限制条件

1. 点击【下一步】离开功能，需确保至少规划一颗螺钉；
2. 进入功能需确保病例功能已经加载X-Ray和CT；
3. 螺钉只能在2D视图上进行规划；
4. 长度测量只能在2D视图上进行；
5. 规划螺钉数量≤50。

## 测试要点

1. 查看2D视图是否正常显示；
2. 手动添加螺钉测试；
3. 螺钉删除测试；
4. 磁盘空间不足，保存数据测试；
5. 螺钉名称、颜色、直径、类型、显示与隐藏修改测试；
6. 点击下一步，数据保存正确性及完整性测试；
7. 点击上一步，返回X-Ray注册页面。

# 术前准备模块

## 模块描述

术前准备是用户在进行手术前对设备和相关配件的进行准备工作，本模块展示了在手术前对不同手术术式的器械推荐摆放位置，以及配准板、定位器等工具的调整使用方法，是核心部分的准备阶段。

## 功能

用户当前处于患者管理模块时，首先选择对应的患者信息后，点击右下角的【下一步】按钮进入术前准备模块。术前准备模块分为工具选择区，术式位置示意图区与设备演示动画区。

术前准备详细操作：

1.软件处于患者管理模块点击【下一步】可进入手术工具选择界面，选择适合本场手术的工具包，C臂机与配准板，点击【确认】进入下一步，点击取消则取消刚才所选，并且返回患者管理模块。

2.进入术前准备界面，选择合适的手术术式以获取设备推荐摆放位置，然后按照演示动画对机械臂与配准板进行安装和摆放，保持执行台车的稳固，固定术中所需的示踪器，完成术前准备操作。

3.完成机械臂的移动，无菌罩，定位器和示踪器的安装，稳固台车后，点击右下角的【下一步】即可进入X片注册模块。

术前准备详细功能：

1. 统计不同工具包的使用次数，提示使用合适的工具包进行手术，载入后续操作所需工具包、C臂机和配准板的相关信息。
2. 提供推荐术式摆放示意图，提供正确的机械臂和台车配置操作示意动画。
3. 记录用户所选择的工具包、C臂机、配准板相关信息，如超越工具包使用寿命并提示用户更换其他可用工具包。

## 性能

无。

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 工具包 | 无 | json | 不为空 | 鼠标点击 | 统计工具包的使用次数 |
| **2** | C臂机 | 无 | json | 不为空 | 鼠标点击 |  |
| **3** | 配准板 | 无 | json | 不为空 | 鼠标点击 |  |

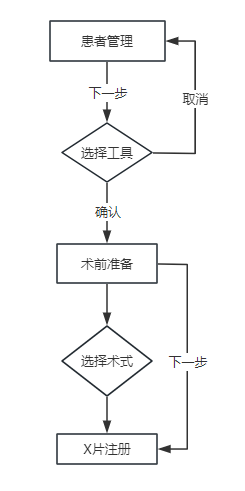
## 输出项目

无。

## 算法

无。

## 程序逻辑



术前准备模块流程图

## 接口

/\*

功能：确认选择的工具包、C臂机和配准板并载入相关信息

\*/

void SureClicked();

/\*

功能：进入术前准备前，先判断有无工具包和C臂机可供使用

\*/

void ToolCArmExits();

/\*

功能：用户点击取消，返回上次用户所选按钮并取消记录操作

\*/

void ReviewBtnStyle();

## 存储分配

术前准备工具选择功能本地文件存储内容：

1. 工具包和C臂机基本信息存储位置目录在：”../../config/system\_config.json”文件；
2. 工具包次数统计目录在：”../../config/system\_config.json”文件与”../../config/tool\_package/”文件夹的子文件中的”tool\_info\_config.json”文件中；
3. C臂机的基本信息存储位置目录在”../../config/c\_arm\_package”下的子文件中的”c\_arm\_info\_config.json”文件中。

## 限制条件

1. 工具包显示数量个数不超过6个；
2. C臂机显示数量个数不超过6个；
3. 配准板只限定机械臂配准板和C臂机配准板；
4. 进入前需要提前导入工具包和C臂机；
5. 工具包和C臂机数量为0不可进入下一步。

## 测试要点

1. 工具包、C臂机、配准板是否成功选择；
2. 切换工具包、C臂机、配准板后是否成功；
3. 选择工具包、C臂机、配准板后取消是否取消成功；
4. 工具包、C臂机、不存在测试；
5. 超过6个数量的工具包、C臂机是否会显示超过6个数量；
6. 患者管理模块能否正确进入术前准备的工具选择界面；
7. X片注册模块能否正确进入术前准备的工具选择界面；
8. 工具包、C臂机、配准板选择后对应的工具包文本是否修改成功；
9. 工具包次数超过50次是否会有提示；
10. 先成功选择工具包、C臂机、配准板后对工具包、C臂机、配准板进行更改再次取消是否取消成功；
11. 工具包、C臂机、配准板选择后点击取消是否复位成功；
12. 工具包次数不同时是否会同步；
13. 切换手术术式和摆放位置示意图是否切换；
14. 动画示意图是否切换后自动播放；
15. 能否成功返回患者管理与成功进入X片注册界面；
16. 术前准备工具选择界面按钮能否选中并且载入工具包和C臂机相关信息。

# X片注册模块

## 模块描述

13.1.1图像接收

1.从成像设备接受DIOCM格式X片数据。

2.以被动接收的方式响应处理数据图像。

13.1.2 图像注册

1.通过硬件结构设计与图像算法以及示踪器数据计算X片图像的空 间坐标姿态。

2.将算法自动识别的结果可视化在图像上，判断算法结果的正确性。

3.在图像算法无法自动识别的情况下，可以手动调整后再重新注册。

13.1.3 图像分组

1.根据手术的目标不同或设备成像范围的限制，需要对X片图像进行分组处理。

2.组间图像数据与注册结果保持分离。

13.1.3 机械臂控制

1.选择工具器械后，在对应的机械臂配准板下。通过调用机械臂控制算法辅助摆放配准板位置。

## 功能

1. 能通过网口接收C臂机传输的X片dicom图像；
2. 接收到图像后自动进行畸变矫正，图像左右镜像，在图像候选区显示，最后接收的图像显示在最左边。
3. 图像候选区域的图像能够任意拖拽到显示窗口，但已经使用的图像不允许重复使用。注册失败的图像不允许使用。
4. 已经注册的图片，在列表中依次展开，无法左右拖动。
5. 图片按照接收顺序进行排序，在图片列表中每个图片左上角有标识。
6. 图片拖拽到显示窗口时，会现在当前显示图片是列表中具体位置，左上角同样有标识。
7. 该界面可以选择切换工具包，选择切换工具包后，当前组的数据全部清空，对于已经定位过的组，不允许切换工具包。
8. 接收X片图像，保存当前患者示踪器和配准板示踪器的数据，并自动进行注册，注册包含标记球的识别和配准.
9. 列表显示小球识别顺序，列表中有标记球编号，标记球二维坐标，和“删除”按钮。能删除对应识别坐标。
10. 放大缩小功能可以对单个坐标周围数据进行缩放，方便取出对应点。
11. 上下左右按钮可以移动坐标点，调整坐标位置。
12. 图像注册精度不足时，可以选择自己调整相应坐标和参数文件进行重新注册，以确保图像注册精度。
13. 注册失败时，语音提示注册失败
14. 机械臂水平、机械臂居中可以根据当前机械臂姿态调整，把机械臂调整成水平或者居中状态，以方便拍摄X片片时，配准板上小球能均匀分布在图像中心位置，减少畸变带来的影响。
15. 可以并存多组数据，数据之间无关联，可以任意切换。

## 性能

注册时间<=10s。

## 输入项目

X注册功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | Xray图片 | 无 | VtkImageData | 不为空 | 通讯功能传入 | X片图片 |
| **2** | Xray内外参 | 无 | QString | 不为空 | 工具包读取 | 内外参数 |
| **3** | 小球三维坐标 | 无 | std::vector<cv::Point3f> | 不为空 | 工具包读取 | 工具参数 |

## 输出项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | XRay图像计算矩阵信息 | 无 | double | 不为空 | 自动保存 | 每张图像内存矩阵\*NDI |
| **2** | XRay正侧位显示结果 | 无 | QString | 不为空 | 自动保存 | 窗口显示信息 |
| **3** | Xray图片是否可用标识 | 无 | Bool | 不为空 | 自动保存 | 是否可用 |
| **4** | Xray路径 | 无 | QString | 不为空 | 自动保存 | 恢复数据 |
| **5** | 当前组是否定位 | 无 | Bool | 不为空 | 自动保存 | 判断新增组 |
| **6** | 组别信息 | 无 | Int | 不为空 | 自动保存 |  |

## 算法

13.6.1 图像矩阵算法：

1. 根据阈值递增从最小阈值到最大阈值分割源图像转换为几张二值图像。
2. 提取连通域分量，通过计算连通域面积、圆度、惯性比、凸度等参数去除不包含目标的区域，返回最终目标的中心位置和连通域半径，此时得到X片图像中检测到的小球的位置。
3. 双层校准板中小球的三维坐标由三坐标仪标定出三维坐标；此时得到2D点集数据和3D点集数据。
4. 为求小球的排列顺序和C臂机外参矩阵，在2D和3D点配准矩阵计算过程中使用如下方法：假设2D点集合中的某一点与3D点集合中的任意点都有对应关系，我们使用两点之间的对齐概率来表示这种关系，通过优化目标函数



来求解变换矩阵R和T，同时通过对齐概率矩阵求得2D和3D点集之间的对应关系。

1. 通过计算出的旋转和平移矩阵将3D点坐标重投影至X片图像上，计算通过算法识别出的2D点的位置和重投影的2D坐标之间的偏差给出算法的重投影误差，即为图像注册精度。

13.6.2 机械臂水平算法

1. 配准板上安装有多个不共面的标记球，标记球由能在成像设备中显影的材料制成。
2. 标记球在机械臂法兰坐标系下的三维坐标通过设计参数或三坐标仪标定获得。
3. 拍摄术中图像，并对图像和配准板的三维坐标进行配准，获得配准板坐标系到图像坐标系的转换关系T.
4. 设定配准板目标姿态Pose，然后根据机械臂当前的姿态数据P即可算出新的机械臂姿态new\_P = P\*inv(T)\*Pose\*T，inv（）表示求逆矩阵。
5. 机械臂根据new\_P运动到位置后，配准板就摆到了我们设定的位置。其中T，P，new\_P和Pose均可表示成4\*4刚体变换矩阵：

其中Pose是图像坐标系中设置的配准板姿态，可设置为任意姿态，例如配准板平面与图像平面水平，配准板平面与机械臂基座平面垂直，或成任意角度等。

13.6.2机械臂居中算法

1. 配准板上安装有多个不共面的标记球，标记球由能在X片和CT中显影的高密度材料制成，例如钢珠球、陶瓷球等。
2. 标记球的三维坐标通过设计参数或三坐标仪标定获得，并通过坐标系转换能够获得标记球在机械臂法兰坐标系下的三维坐标。
3. 在配准板上建立一个坐标系A，选择三个点建立一个平面，确定了平面的法向量，设定为XYZ三维坐标系的其中第一轴。再根据上述三个点中的其中两个点确定一个单位向量，设定为第二轴。第三轴通过第一轴叉乘第二轴获得。再选定任意一个点作为坐标轴的原点。建立好坐标系后，就得到了坐标系A和法兰坐标系间的转换关系M。、
4. 根据机械臂当前的姿态数据P和M即可算出当前配准板在机械臂基座坐标系下的位置(x0, y0, z0)和姿态。
5. 可以设定配准板目标姿态Pose，然后根据机械臂当前的姿态数据P和M即可算出新的机械臂姿态new\_P = Pose\*inv(M)，inv()表示求逆矩阵
6. P，new\_P和Pose均可表示成4\*4刚体变换矩阵：

(x0, y0, z0)的计算如下：

=P\*M

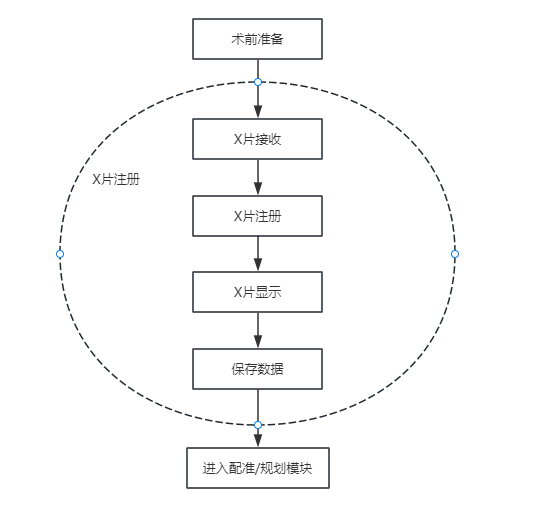
其中Pose是坐标系A相对于机械臂基座坐标系的刚体变换矩阵，其中左上角3\*3的旋转矩阵可由旋转角、旋转向量、四元数转化而来，位置偏移量为（x0, y0, z0）。

机械臂根据new\_P运动到位置后，配准板就摆到了我们设定的位置。

1. 根据Pose的不同，配准板可以摆出任意位置，例如配准板平面与机械臂基座平面水平，配准板平面与机械臂基座平面垂直，或成任意角度。

## 程序逻辑

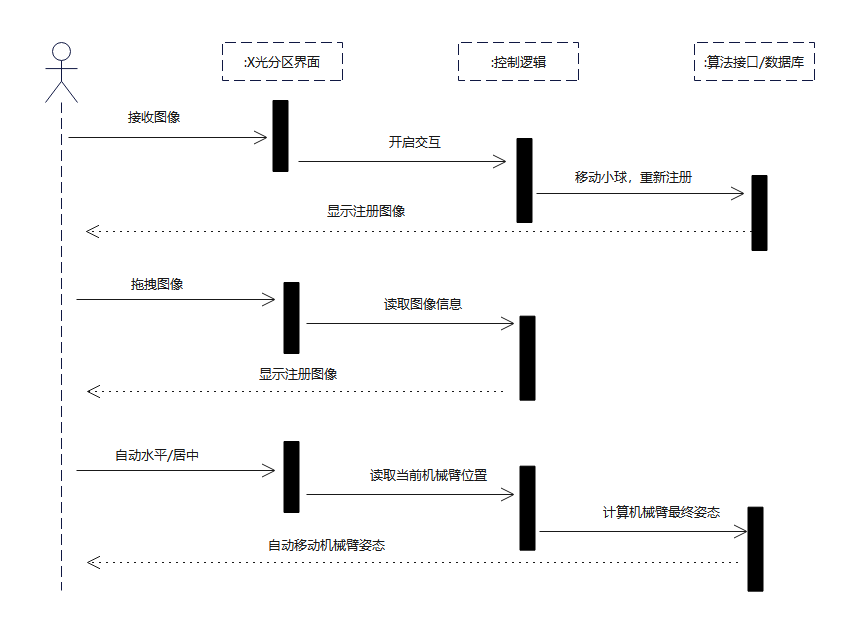
X片注册功能是本软件顺序流程的一个步骤，上一步为术前准备，下一步为规划功能。分别用X片注册功能在软件内的流程图、X片注册功能系统用例图、X片注册功能时序图来展示本功能的程序逻辑。



X片注册功能在软件内的流程图



X片注册功能系统用例图



X片注册功能时序图

## 接口

X片注册功能包含XrayRegisterWidget、 XrayRegisterDialog、tablewidget\_updown、DicomStoreImageService四个主要类。其中重要接口详细说明如下：

* + 1. **XrayRegisterWidget**

/\*

\* 功能描述：初始化恢复图像数据到内存

\*/

void InitRecoverDatatoRam();

/\*

\* 功能描述： 初始化显示数据

\*/

void InitShowRecoverData();

/\*

\* 功能描述： 恢复图像数据

\* 参数0：图像路径

\* 参数1：是否可用标识

\* 参数2：图像label存入list

\*/

void RecoverDcmData(QString path, double usable, QList<QLabel\*>& temp\_label\_);

/\*

\* 功能描述： 计算当前界面的显示数据

\* 参数0：拖拽图像所在位置

\* 参数1：对应窗口编号

\*/

void Caculate\_Window\_show(int row, int widget\_index);

/\*

\* 功能描述： 计算NDI\患者\actor和图像矩阵

\* 参数0：图像actor矩阵

\* 参数1：相机矩阵

\*/

void Caculate\_Matrix\_Cam\_(vtkSmartPointer<vtkMatrix4x4> actor\_Matrix, vtkSmartPointer<vtkMatrix4x4> Cam\_Matrix);

/\*

\* 功能描述：创建新组

\* 参数0：新组id号

\* 参数1：构造group对象

\*/

void CreatNewGroup(int new\_index\_, XrayGroupInfoJson\* group);

/\*

\* 功能描述：切换分组

\* 参数0：切换组id号

\*/

void Switch\_Group\_ById(int group\_id);

/\*

\* 功能描述：清除内存所有数据

\*/

void ClearAllRamData();

/\*

\* 功能描述： 图像接收处理

\*/

void DcmProcess();

/\*

\* 功能描述： 切换工具

\*/

void ChangeTools();

/\*

\* 功能描述： 切换病人

\*/

void PatientChanged();

/\*

\* 功能描述： 机械臂水平

\*/

void on\_horizontal\_button\_pressed();

* + 1. **XrayRegisterDialog**

/\*

\* 功能描述： 初始化视图

\*/

void InitView();

/\*

\* 功能描述： 初始化列表

\*/

void InitLabel();

/\*

\* 功能描述： 创建点标签

\*/

void CreatePointLabel();

/\*

\* 功能描述： 创建数字标签

\*/

void CreateNumLabel();

/\*

\* 功能描述： 缩放图像

\*/

void ResizeImage();

/\*

\* 功能描述： 数据存入table

\*/

void AddTableData();

/\*

\* 功能描述： 删除lable

\*/

void RemoveLabel();

/\*

\* 功能描述： 更新table数据

\*/

void updateTableData();

/\*

\* 功能描述： 移动label

\*/

void MoveLabel();

/\*

\* 功能描述： 计算机械臂位置

\*/

void CaculateTcpCenterPos();

/\*

\* 功能描述： 计算图像矩阵\NDI矩阵关系

\* 参数0：NDI下患者矩阵

\* 参数1：NDI下C臂机或者机械臂矩阵

\* 参数2：图像矩阵

\* 参数3：图像actor矩阵

\* 参数4：相机矩阵

\*/

void CaculateCameraMatrix(vtkSmartPointer<vtkMatrix4x4> actMat\_Hz, vtkSmartPointer<vtkMatrix4x4> actMat\_Cb, vtkSmartPointer<vtkMatrix4x4> actMat\_Im, vtkSmartPointer<vtkMatrix4x4> actMat\_Render, vtkSmartPointer<vtkMatrix4x4> actMat\_CamMat);

/\*

\* 功能描述：放大图像

\*/

void ScaleBigImg();

/\*

\* 功能描述：缩小图像

\*/

void ScaleBigImg();

* + 1. **tablewidget\_updown**

/\*

\* 功能描述： 删除数据

\*/

void MoveData();

/\*

\* 功能描述：更新数据

\*/

void updateData();

/\*

\* 功能描述： 增加数据

\*/

void AddData();

/\*

\* 功能描述： 增加删除按钮

\*/

void addBtn();

* + 1. **DicomStoreImageService**

/\*

\* 功能描述： 设置传输Title

\* 参数0：设置与C臂机通信的title

\*/

void setAETitle(const QByteArray& aeTitle);

/\*

\* 功能描述：设置端口

\* 参数0：设置与C臂机通信的端口

\*/

void setPort(int port);

/\*

\* 功能描述：设置图像保存路径

\* 参数0：图像保存路径地址

\*/

void setDicPath(std::string dicPath);

## 存储分配

接收X片图片信息存在本地文件夹“./data/当前病例/Xray/”中

X片的相关信息和图像显示信息存在“operation\_data\_json.json”文件内。

## 限制条件

X片组数不超过5组。

数据不能重复使用。

## 测试要点

图像接收是否正常；

机械臂水平居中是否可用运动；

小球坐标识别是否正常，是否可以移动、删除坐标；

放大缩小功能是否可以使用。

分组数据是否互相没有关联。

切换工具时数据是否被清空。

切换病人时数据是否更新位当前病人数据。

数据恢复是否可以正常恢复。

# 图像配准模块

## **模块描述**

1. 配准功能是本软件实现定位功能的重要途径，是软件的重要功能。目的是使已经导入的CT数据和已经注册的X-Ray图像进行配准，从而达到统一坐标系的效果；
2. 配准功能在软件中属于常驻内存、不可重入，属于顺序处理且代码没有覆盖要求。本功能与其他功能耦合较小：利用单例的静态指针获取CT数据和X-Ray数据，同自动配准或手动配准将CT与X-Ray的位置对应起来。
3. 功能内部代码增加边缘性检测，详见接口部分。

## **15.2. 功能**

1. 图像叠加显示X片图像数据和透明模式的三维CT图像数据。
2. 可单独调整X片与CT图像的窗宽窗位，X片图像的窗宽窗位点击“窗宽窗位”按钮出现滑条进行调节，默认窗宽窗位与上一界面的相同，CT图像的窗宽窗位依靠鼠标右键调节。
3. 鼠标左键可对三维CT图像旋转，鼠标滚轮对CT图像放大缩小，按住滚轮移动鼠标可平移CT图像
4. 颜色选择按钮预设10种颜色，选择后CT图像变成相应的颜色，默认为浅蓝色。
5. 初次打开界面，CT与X片数据进行粗配准。
6. 非首次打开界面，默认显示上一次的配准后的结果。
7. 还原功能把配准结果还原到初配状态。显示CT的透明度也还原到初始值。
8. 有调节CT和X片图像显示比例的滑动条，播放按钮，点击后可自动循环切换显示比例。
9. 通过鼠标和按钮操作，可以对CT图像进行平移和旋转，手动将CT和X片图像大致对齐，通过观察两个图像差异判断配准情况。
10. 点击配准按钮，对CT与X片图像进行自动配准，在左右两组图像框选封闭的矩形可以对图像区域进行配准，未框选的情况下按照整张图像进行配准。
11. 播放按钮，自动设置不同的亮度和透明度以展示CT数据。
12. 左右两边分别一个滑条，也可以手动调节CT的亮度和透明度。
13. 上下左右、X轴方向转、Y轴方向转、Z轴方向转、放大缩小用于调整CT数据，方标后续配准。
14. 保存配准结果矩阵。

## **性能**

1. 三维数据重建速度≥15MPS。

## **输入项目**

配准功能-全功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | Orth2DModel | 无 | 指针 | 不为空 | 自动输入 | CT数据和X-Ray数据信息 |

## **输出项目**

配准功能-全功能输出项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | Orth2DModel | 无 | 指针 | 不为空 | 自动保存 | 配准矩阵 |

## **算法**

**15.6.1. 配准算法**

1.算法目的：

将三维CT图像与二维X片图象配准，通过对CT三维数据DRR重建后的二维图像与X片二维图像求图像相关性，搜索相关性最高的结果作为配准结果。

2.算法流程：

1）确定搜索范围T，最大步长S=T，确定收敛步长S<0.1；

2）以步长S向五个自由度正反方向搜索（正位图像对前后方向不敏感），计算正位相关性；

3）计算搜索过程中相关性最大的值与当前位置相关性进行比较；

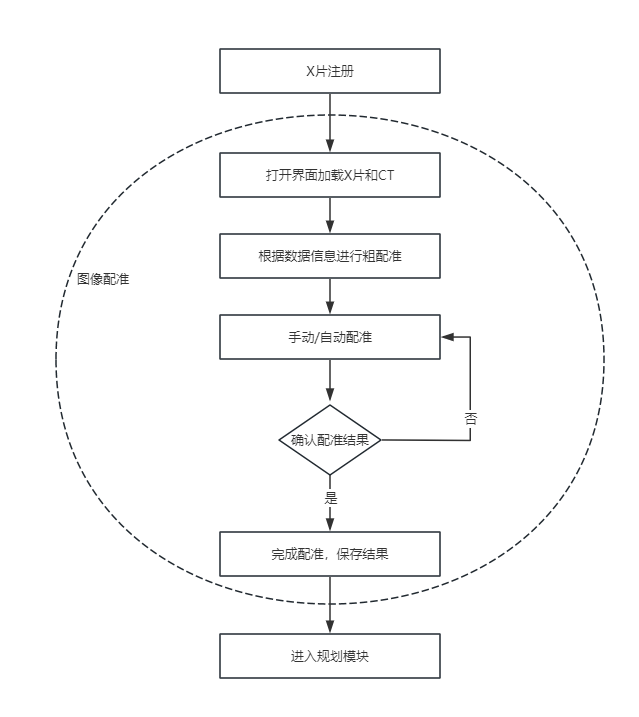
4）如果存在相似度比当前位置相似度高，更新当前位置。如果不存在令S=S/2。重复2，3，4；

5）当S<0.1,令S=T。以步长S向前后方向搜索，计算侧位相关性；

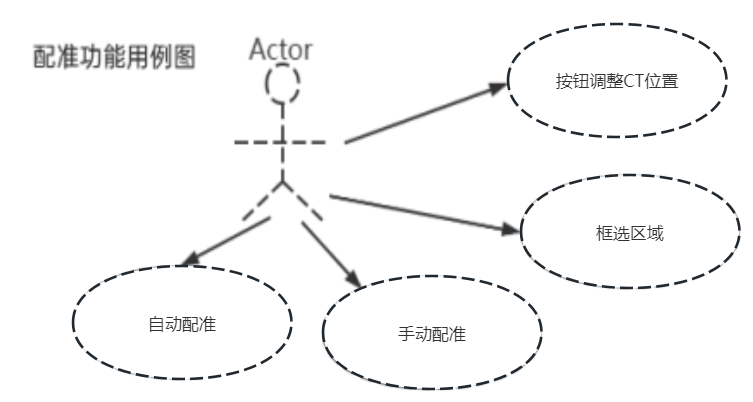
6）与3、4类似，直到S<0.1跳出搜索。结果即作为配准结果。

## **15.7. 程序逻辑**

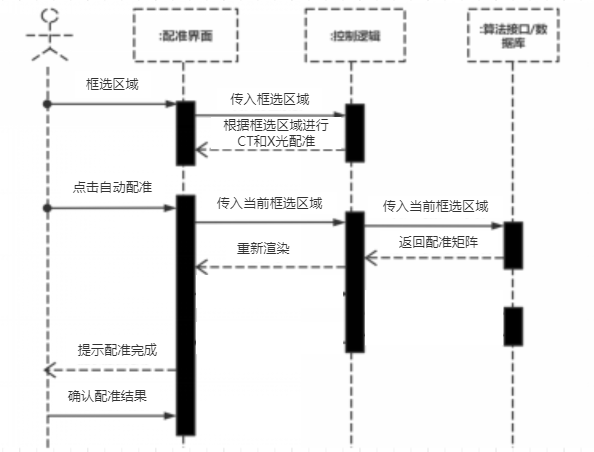
配准功能是本软件顺序流程的一个步骤，上一步为X片注册，下一步为规划功能。分别用配准功能在软件内的流程图、配准功能系统用例图、配准功能时序图来展示本功能的程序逻辑。



配准功能在软件内的流程图



配准功能系统用例图



配准功能时序图

## **15.8. 接口**

配准功能包含ImageRegistrationWidget、SingleXrayShowWidget两个主要类。其中重要接口详细说明如下：

**12.8.1.** ImageRegistrationWidget

/\*

\*功能描述: 框选区域

\* 参数0：x轴

\* 参数1：y轴

\* 参数2：宽带

\* 参数3：高度

\* 参数4：具体窗口

\*/

void DrawRectPaint(int x, int y, int w, int h, int widget\_index);

/\*

\* 功能描述： 更新窗宽窗位

\* 参数1：窗宽

\* 参数2：窗位

\* 返回值：无

\*/

void UpdateWinLevel(int c\_x, int c\_y)

/\*

\* 功能描述： 选择并改变3d颜色

\* 参数0：颜色id

\* 参数1：是否选择

\* 返回值：无

\*/

void SlotSelectedNewColor(int id, bool checked)

/\*

\* 功能描述： 还原功能

\* 参数：无

\* 返回值：无

\*/

void on\_reset\_button\_clicked();

/\*

\*功能描述: 粗配准

\*/

void CoarseMatching();

/\*

\*功能描述: 粗配准矩阵

\*/

void InitXmatMatrix();

/\*

\*功能描述: CT移动

\* 参数0：移动步数

\* 参数1：具体方向

\*/

void CTMove(double step, int i);

/\*

\*功能描述: 设置颜色

\* 参数0：颜色

\*/

void setColor(QColor& c);

/\*

\*功能描述:初始化框选窗口数据

\*/

void InitDrawRect();

/\*

\*功能描述:清空框选窗口数据

\*/

void RemoveAllRect();

/\*

\*功能描述:初始化滑条

\*/

void InitJQtSlider(WidgetSkin skin);

/\*

\*功能描述:保存配准数据

\*/

void SaveMat2Json();

/\*

\*功能描述: 框选配准区域

\*/

void on\_select\_button\_clicked();

/\*

\*功能描述: 关闭界面

\*/

void on\_nextstepBtn\_clicked();

/\*

\*功能描述: 还原功能

\*/

void on\_reset\_button\_clicked();

/\*

\*功能描述: 还原功能

\*/

void on\_color\_button\_clicked();

/\*

\*功能描述: 配准

\*/

void on\_regist\_button\_clicked();

/\*

\*功能描述: x轴向上旋转

\*/

void on\_x\_up\_rotate\_btn\_clicked();

/\*

\*功能描述: x轴向下旋转

\*/

void on\_x\_down\_rotate\_btn\_clicked();

/\*

\*功能描述: y轴向上旋转

\*/

void on\_y\_up\_rotate\_btn\_clicked();

/\*

\*功能描述: y轴向下旋转

\*/

void on\_y\_down\_rotate\_btn\_clicked();

/\*

\*功能描述: z轴向上旋转

\*/

void on\_z\_up\_rotate\_btn\_clicked();

/\*

\*功能描述: z轴向上旋转

\*/

void on\_z\_down\_rotate\_btn\_clicked();

/\*

\*功能描述: 放大功能

\*/

void on\_scale\_big\_btn\_clicked();

/\*

\*功能描述: 缩小功能

\*/

void on\_scale\_small\_btn\_clicked();

/\*

\*功能描述: 向上移动

\*/

void on\_UpButton\_clicked();

/\*

\*功能描述: 向下移动

\*/

void on\_DownButton\_clicked();

/\*

\*功能描述: 向左移动

\*/

void on\_LeftButton\_clicked();

/\*

\*功能描述: 向右移动

\*/

void on\_RightButton\_clicked();

/\*

\*功能描述: 下面滚动事件

\* 参数0：当前位置

\*/

void on\_left\_table\_scroll\_bar\_changed(int value);

/\*

\*功能描述: 下面滚动事件

\* 参数0：当前位置

\*/

void on\_right\_table\_scroll\_bar\_changed(int value);

/\*

\*功能描述: 移动滑动条

\*/

void MoveSlider();

**12.8.2.** SingleXrayShowWidget

/\*

\* 功能描述：载入X片

\* 参数0：X-Ray vtkImageData

\* 参数1：cam矩阵

\* 参数2：actor矩阵

\* 返回值：无

\*/

void LoadXRay(vtkSmartPointer<vtkImageData> imgData, vtkSmartPointer<vtkMatrix4x4> camMat, vtkSmartPointer<vtkMatrix4x4> cb2cam);

/\*

\* 功能描述：载入X片

\* 参数0：X-Ray cv::Mat

\* 参数1：cam矩阵

\* 参数2：actor矩阵

\* 返回值：无

\*/

void LoadXRay(cv::Mat cvMatImg, vtkSmartPointer<vtkMatrix4x4> camMat, vtkSmartPointer<vtkMatrix4x4> cb2cam);

/\*

\* 功能描述：获得vtkRender

\* 参数0：目标渲染层

\* 返回值：无

\*/

void GetVtkRender(vtkSmartPointer<vtkOpenGLRenderer> &targetRender);

/\*

\* 功能描述：更新按钮状态

\* 参数0：按钮状态

\* 返回值：无

\*/

bool UpdateButtonStatus(QPushButton\* btn);

/\*

\* 功能描述：更新按钮状态

\* 参数0：按钮对象

\* 参数1：目标状态

\* 返回值：无

\*/

void UpdateButtonStatus(QPushButton\* btn, bool status);

/\*

\* 功能描述：更新状况窗位按钮状态

\* 参数0：目标状态

\* 返回值：无

\*/

void UpdateWWWLButtonStatus(bool status);

/\*

\* 功能描述：处理当前图像

\* \*参数：

\* 返回值：无

\*/

void HandleCurrentImage();

/\*

\* 功能描述：切换按钮状态

\* 参数0：按钮对象

\* 返回值：无

\*/

void SwitchButtonStatus(QPushButton\* btn);

/\*

\* 功能描述：刷新图像

\* 返回值：无

\*/

void RefreshImage();

/\*

\* 功能描述：改变背景对比度

\* 参数0：左滑头位置

\* 参数1：右滑头位置

\* 返回值：无

\*/

void ChangeBackgroundContrast(int left, int right);

/\*

\* 功能描述：判断是否载入图片

\* 返回值：无

\*/

bool IsImageNull();

/\*

\* 功能描述：获得交互器

\* 参数0：目标交互器

\* 返回值：无

\*/

void GetInteractor(vtkSmartPointer<vtkRenderWindowInteractor>& targetInteractor);

/\*

\* 功能描述：获得通道位置

\* 参数0：左滑头位置

\* 参数1：右滑头位置

\* 返回值：无

\*/

void GetLinePos(double\* point1, double\* point2);

/\*

\* 功能描述：获得按钮状态

\* 返回值：按钮状态

\*/

int\* GetButtonStatus();

/\*

\* 功能描述：载入按钮状态

\* 参数0：载入按钮状态

\* 返回值：无

\*/

void LoadButtonStatus(int\* buttonStatus);

/\*

\* 功能描述：获得窗口激活状态

\* 返回值：窗口序号

\*/

int GetWidgetActivatedState();

/\*

\* 功能描述：设置渲染窗口选中状态

\* 返回值：无

\*/

void SetWidgetActivatedState(bool state);

/\*

\* 功能描述：获取相机矩阵

\* 返回值：无

\*/

void GetCameraMat(vtkSmartPointer<vtkMatrix4x4> CameraMat);

## **15.9. 存储分配**

配准功能本地文件存储内容：无

## **15.10. 限制条件**

1. 确认配准结果后才能关闭界面；

## **15.11. 测试要点**

1. 加载数据是否准确；

2. 配准后还原，查看是否返回原始姿态；

3. 多次配准查看结果是否变化；

4. CT颜色修改，窗宽窗位调整及重置测试；

5. X片片小球显示与隐藏、滤波开启与关闭测试；

6. CT与X片对比度调节；

7. 正位或侧位分别选中微调；

8. 直接点下一步，有提示；

9. 自动配准后，点下一步，数据保存正确性验证；

10.框选区域后的配准结果验证；

# 定位导航模块

## 模块描述

1. 定位导航模块是模拟并定位机械臂到选中的规划螺钉的实际位置的模块。定位导航模块会根据患者示踪器和套筒示踪器信息，螺钉信息计算出机械臂末端的目标位姿，仿真定位到这个位置。
2. 模块展示了冠状，矢状，横断面的切片信息,规划的螺钉和螺钉列表,螺钉的出入点偏差和工具的出入点偏差。
3. 可对评估界面查看，末端控制界面可选中用脚踏控制或界面控制机械臂的仿真定位，沿通道上下移动，旋转。

## 功能

1. 点击右上角的评估按钮，弹窗评估界面查看效果。
2. 进入定位界面时弹框提示“请确认升降脚撑已落地”并图片示意，点击“确定”或“X”按钮后关闭弹框。
3. 在定位导航的螺钉列表中，点击单个螺钉的螺帽显示隐藏按钮可以对螺帽显示隐藏,点击所有螺钉的螺帽显示隐藏按钮可以对所有螺钉螺帽显示隐藏。点击单个螺钉的显示隐藏按钮可以显示隐藏当前螺钉，点击所有螺钉的显示隐藏按钮可对所有螺钉显示隐藏。
4. 点击末端控制按钮，可对末端控制开关状态切换。末端控制关闭时，机械臂末端按钮无效，脚踏定位无效。可通过末端控制界面的仿真和定位按钮，对机械臂仿真定位操作。机械臂定位到达速度阈和角度阈后,沿通道上下移动，旋转按钮启用，长按可对机械臂末端沿通道上下移动，旋转。末端控制打开时，可通过踩下执行脚踏，对机械臂进行仿真定位操作，定位到达速度阈和角度阈后，同时按下机械臂末端按键和踩下执行脚踏，可对机械臂末端沿通道上下移动，旋转。

定位精度不足时，会弹窗提示重新定位。示踪器不可见时，会弹窗提示移动NDI让示踪器可见。

1. 定位过程中，会显示通道与螺钉，工具与螺钉的入点出点偏差，在精度范围内用绿色显示，精度范围外用黄色显示，示踪器不可见，用黑色N/A显示。
2. 仿真界面为弹框，包含机械臂姿态显示区域，角度滑动条，高度滑动条，“仿真”，“确认”、“关闭”按钮。并显示当前螺钉名称。
3. 选择螺钉有语音播报螺钉名称。
4. “末端控制器”关闭时，点击“仿真”按钮，出现弹框，根据螺钉数据、套筒示踪器、患者示踪器的位姿数据、推荐的高度和角度计算定位姿态和运动路径，并在视图中显示。
5. 可通过滑动条调节高度和角度后再点击“仿真”，对运动路径和定位姿态进行计算，并显示。高度范围50mm-200mm，默认100mm。角度范围-180°-180°，默认0°
6. “末端控制器”开启时，“仿真”、“定位”、“↑”、“↓”、“左旋”、“右旋”按钮均置灰，用末端控制器的“↑”、“↓”按钮选择螺钉，踩下自动脚踏进入仿真界面，自动进行仿真。
7. 末端控制器权限开关，默认为开启状态，点击后为关闭状态（此时末端控制器不起作用）
8. 确定当前仿真结果后，点击“确认”按钮，弹窗关闭。
9. 具备可视化机械臂定位仿真功能，确定螺钉后根据目标位置姿态提供可视化仿真，允许一定程度修改目标机械臂姿态。
10. 通过机械臂控制箱读取硬件脚踏、按钮信号。
11. 精度不足时，弹框提示“精度不足，请再次定位”并有语音提示，点击“X”关闭弹框，踩下自动脚踏，机械臂继续运动，直到精度小于阈值。
12. 机械臂运动过程中，示踪器被遮挡法兰灯光变为黄色；未运动时，示踪器被遮挡法兰灯光颜色不变。
13. 运动到目标位置且误差小于阈值法兰灯光状态改变（由蓝色变绿色）。
14. 定位过程中或定位完成时，机械臂出现故障，弹框提示故障信息，点击弹框中的“故障解除”按钮一键解除故障，故障解除后踩下脚踏继续运动至精度达标位置。
15. 如果机械臂掉电并与主机断开连接，进入机械臂控制界面，重新启动和连接机械臂，返回定位界面再次点击“仿真”和“定位”，踩下脚踏继续执行。
16. 末端控制器权限开关，默认为开启状态，点击后为关闭状态（此时末端控制器不起作用）。
17. 定位步骤实时显示定位精度，精度数字通过内容与颜色区分跟踪状态。在系统精度范围内允许定位通道沿目标通道上下移动以及左右旋转。
18. 工具名称会根据当前识别的工具进行改变，“探针”或“骨钻”。如果同时识别到骨钻示踪器和探针，图像中只绘制骨钻，精度只显示骨钻精度。
19. 支持工具实时导航，包括探针和骨钻。同时只跟踪一个工具，根据工具识别情况选择跟踪。实时显示工具导航精度。精度数字通过内容与颜色区分跟踪状态。
20. 完成标定后，识别到工具示踪器和患者示踪器，根据转换关系将工具绘制在二维图像中。
21. 定位出现异常，需给出相应提示。过程中控制灯光指示用户定位状态。
22. 点击返回按钮，切换到上一步骤,点击完成按钮选择结束手术或返回X片注册界面。

## 性能

仿真计算时间<=3s。

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | 高度 | screw\_height | Double | 50-200 | 滑动条 |  |
| 2 | 角度 | screw\_angle | Double | -180-180 | 滑动条 |  |

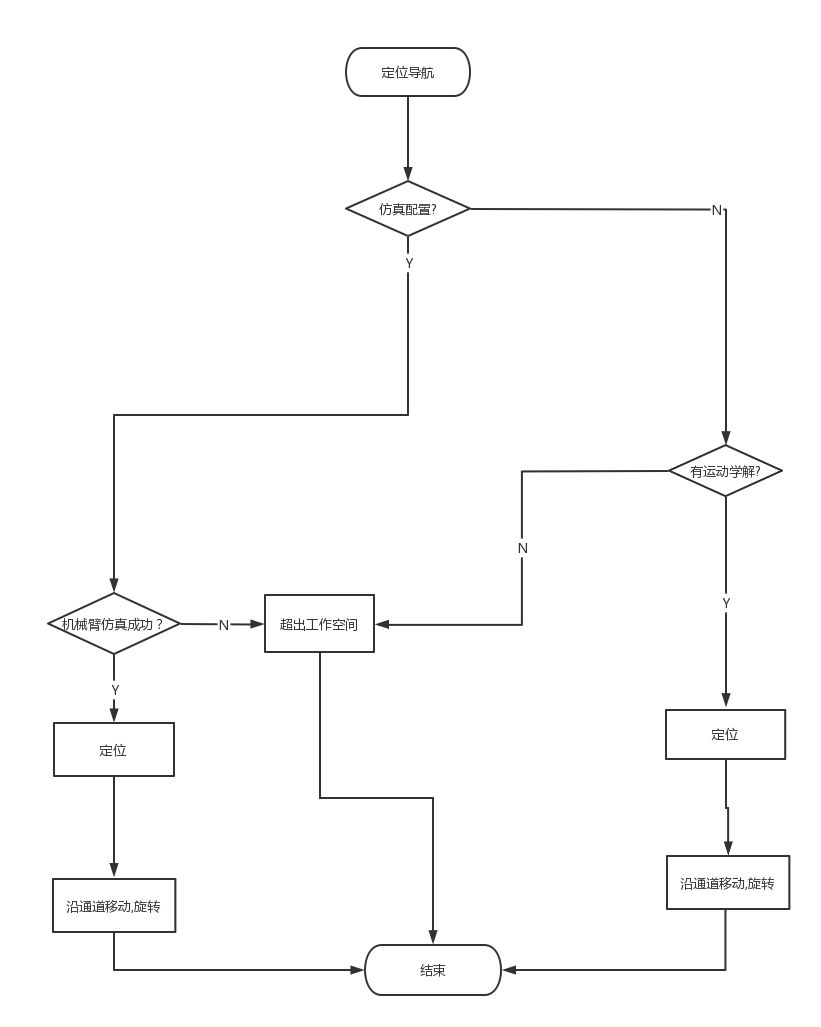
## 输出项目

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **标识** | **数据类型** | **有效范围** |
| 通道螺钉入点偏差 | dis\_screw\_enter\_ | double |  |
| 通道螺钉出点偏差 | dis\_screw\_out\_ | double |  |
| 工具螺钉入点偏差 | tool\_dis\_enter | double |  |
| 工具螺钉出点偏差 | tool\_dis\_out | double |  |

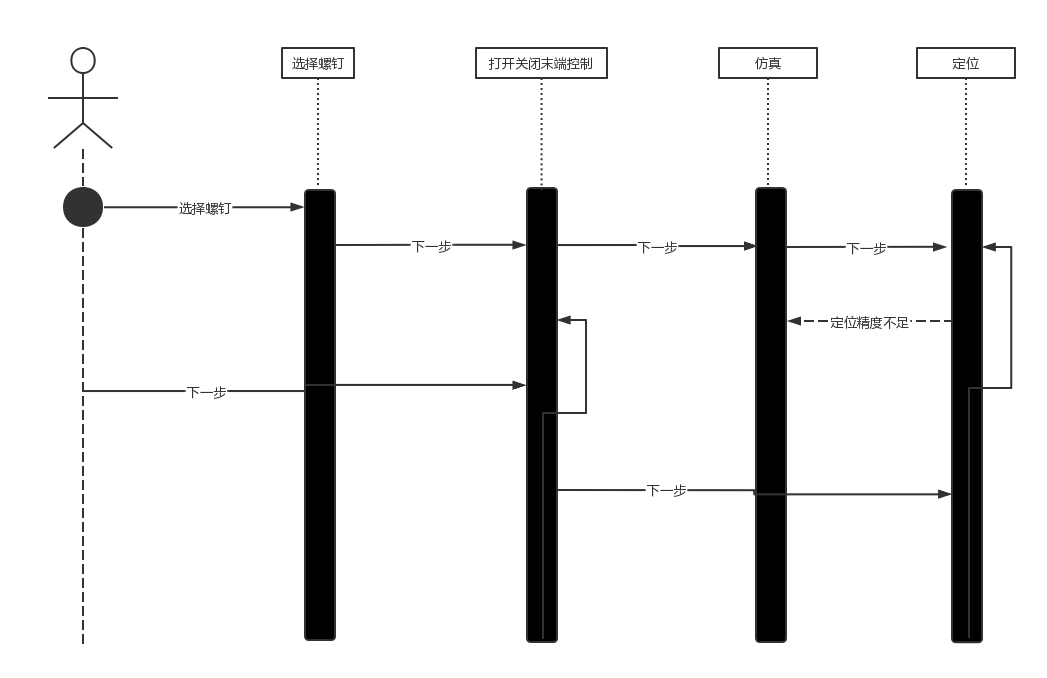
## 算法

无

## 程序逻辑



定位导航流程图



定位导航时序图

## 接口

定位导航模块主要包含ArmSimulationWidget，ScrewTableView，Location2DWidget类。主要的接口如下:

### ArmSimulationWidget

/\*

\*功能描述:滑动条高度角度值设置

\*参 数:

\*\*offset:高度

\*\*angle:角度

\* 返回值:void

\*/

void SetOffsetAndAngle(double offset, double angle);

/\*

\*功能描述:计算定位目标位置关节角

\*参 数:

\*\*joint\_angle\_vec:目标位置关节角

\* 返回值:bool:true:成功 false:失败

\*/

bool CalculateTargetPos(std::vector<double>& joint\_angle\_vec);

/\*

\*功能描述:计算仿真各关节单步转过角度

\*参 数:无

\* 返回值:bool:true:成功 false:失败

\*/

bool CalculteStepMoveAngle();

/\*

\*功能描述:仿真按钮按下

\*参 数:无

\* 返回值:void

\*/

void on\_simulate\_push\_button\_clicked();

/\*

\*功能描述:机械臂仿真模型移动到目标位置

\*参 数:无

\* 返回值:void

\*/

void FreshSimulateTargetAssembly();

/\*

\*功能描述:机械臂仿真模型移动到当前位置

\*参 数:无

\* 返回值:void

\*/

void FreshActualAssembly();

### ScrewTableView

/\*

\*功能描述:指定列锁定

\*参 数:无

\* 返回值:void

\*/

void ScrewTableLocker();

/\*

\*功能描述:指定列解除锁定

\*参 数:无

\* 返回值:void

\*/

void ScrewTableUnLocker();

/\*

\*功能描述:根据螺钉列表类型，隐藏显示指定列

\*参 数:screw\_view\_type:螺钉列表类型

\* 返回值:void

\*/

void SetScrewViewType(const ScrewTableViewType screw\_view\_type = ScrewTableViewType::SCREW\_VIEW\_COMPLETE);

/\*

\*功能描述:给螺钉列表添加一行数据

\*参 数:无

\* 返回值:void

\*/

void AddScrewItem();

/\*

\*功能描述:更新指定类型的表头视图

\*参 数:column\_type:表头类型

\* 返回值:void

\*/

void SlotUpdateCustomHeaderView(ScrewColumnType column\_type);

/\*

\*功能描述:寻找当前行中最接近螺钉列表长度的下拉框数值

\*参 数:screw\_id:螺钉id

\* screw\_length:螺钉长度

\* 返回值:void

\*/

void SlotUpdateScrewLength(int screw\_id, double screw\_length);

### Location2DWidget

/\*

\*功能描述:螺钉切换时,设置显示螺钉名称，语言播报

\*参 数:

\*screw\_name\_value:选中的螺钉名称

\* 返回值:void

\*/

void SlotScrewValueChanged(const QString& screw\_name\_value);

/\*

\*功能描述:末端按键螺钉切换

\*参 数:

\*handle\_direction:方向按键

\* 返回值:void

\*/

void SlotHandleSwitchScrew(HandleDirection handle\_direction);

/\*

\*功能描述:开始定位

\*参 数:无

\* 返回值:void

\*/

void on\_location\_push\_button\_pressed();//定位按钮按下

/\*

\*功能描述:结束定位

\*参 数:无

\* 返回值:void

\*/

void on\_location\_push\_button\_released();//定位按钮松开

/\*

\*功能描述:仿真按钮按下

\*参 数:无

\* 返回值:void

\*/

void on\_simlate\_push\_button\_clicked();//仿真按钮

/\*

\*功能描述:切换螺钉

\*参 数:

\*\*model\_index:当前切换螺钉的数据模型标签

\* 返回值:void

\*/

void ChooseRowItem(const QModelIndex& model\_index);

/\*

\*功能描述:获取nailup,naildown的空间坐标

\*参 数:无

\* 返回值:void

\*/

void SlotSetScrewNailUpDown();

/\*

\*功能描述:语音播报

\*参 数:

\*text:语音播报文字

\* 返回值:void

\*/

void SetTextToSpeech(const QString& text);

/\*

\*功能描述:更新界面显示的出入点偏差

\*参 数:无

\* 返回值:void

\*/

void UpdateScrewEnterOutOffset();//更新出入点偏差

/\*

\*功能描述:更新探针，骨钻偏差

\*参 数:无

\* 返回值:void

\*/

void UpdateTzGzOffset();

/\*

\*功能描述:更新脚踏踩下，仿真定位状态

\*参 数:无

\* 返回值:void

\*/

void UpdateFootSwich();

/\*

\*功能描述:脚踏踩下，仿真定位

\*参 数:

\*b\_foot\_swich:脚踏踩下标志

\* 返回值:void

\*/

void SpineSimlauteAndLocation(bool b\_foot\_swich);

/\*

\*功能描述:获取nailup,naildown空间坐标

\*参 数:

\*nail\_up:螺钉上沿

\*nail\_down:螺钉下沿

\* 返回值:void

\*/

void GetNailupNailDown(double\* nail\_up, double\* nail\_down);

/\*

\*功能描述:检查出入点偏差是否在阈值范围内

\*参 数:无

\* 返回值:bool:true:阈值内 false:阈值外

\*/

bool GetSpineRobotDistanceError();

/\*

\*功能描述:检查速度阈角度阈偏差是否在阈值范围内

\*参 数:无

\* 返回值:bool:true:阈值内 false:阈值外

\*/

bool GetTcpError();

## 存储分配

无

## 限制条件

1. 定位精度阈值为1.5mm
2. 速度阈值为40,角度阈值为6

## 测试要点

1. 螺钉和螺钉列表选项显示是否一致
2. 按钮控制仿真定位和脚踏控制仿真定位是否互斥
3. 定位和仿真位姿是否一致
4. 通道螺钉出入点偏差是否正确
5. 工具螺钉出入点偏差是否正确
6. 机械臂末端灯环颜色显示是否正确

# 反馈验证模块

## 模块描述

在定位界面，点击评估按钮可显示反馈界面。反馈界面包含规划图像显示窗口、对比图像显示窗口、螺钉显示状态按钮、图像切换按钮。反馈界面主要用于显示术前规划X-Ray和术后X-Ray，用以查看实际螺钉位置与术前规划螺钉位置的差异。



二维反馈功能IPO图

## 功能

1. 反馈界面包含两个窗口，左侧窗口显示规划用的X片图像及其对应的规划螺钉，右侧窗口显示新接收的置钉后的X片图像。
2. 窗口显示图像可以翻页切换，每个窗口对应有两个翻页键（上页键、下页键），点击翻页键用以显示图像，更新并显示当前图像的序号，循环翻页。左显示窗口对应的两个翻页键用以切换规划的X片，右侧显示的窗口对应的两个翻页键用以切换术后的置钉的X片。
3. 右侧的对比图像显示窗口显示新发送的X片图像，新接收图像覆盖原有图像显示，并显示当前图像的序号。接收图片检查磁盘空间，如果磁盘不足会拒接收发过来的图像并给出提示。只要磁盘空间充足，接收的图片数量无上限。
4. 界面上有”螺钉显示/隐藏”按钮，和”螺钉样式”按钮，”螺钉显示/隐藏”按钮用以控制隐藏/显示左侧的规划X片显示窗口里螺钉的隐藏/显示，”螺钉样式”按钮用以控制左侧的规划X片显示窗口里螺钉以螺钉样式/圆柱样式显示。
5. 左右两个显示窗口里有X片图像处理栏，默认收起以“图像处理栏”按钮显示，点击“图像处理栏”按钮图像处理栏展开，展开里面是“图像旋转”按钮、“图像还原”按钮、“图像对比度调解”按钮、“图像去小球”按钮、“图片增强”按钮、“图片反色”按钮和“图像处理栏收缩”按钮。这些图像处理按钮对应的图像处理功能全部可作用于当前显示窗口里的X片，点击“图像还原”按钮，X片可被还原。
6. 点击“关闭”回到定位导航界面。再次返回评估界面，之前的图像依然存在。

## 性能

1.反馈验证模块无性能要求。

## 输入项目

反馈-全功能输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 保存的dcm信息 | 无 | FMODEL(Orth2DModel)->operation\_data\_json\_ | 不为空 | 自动输入 | 内存中获取 |
| **2** | 通讯模块保存的X-Ray数据 | 无 | MainModel::sendDcmPath | 已建立连接 | 自动输入 | 接收通讯模块发过来的数据 |
| **3** | 规划模块已经加载的X-Ray和规划通道信息 | 无 | FMODEL(Orth2DModel) | 不为空 | 自动输入 | 规划时用到的数据 |

## 输出项目

反馈-全功能输出项目如下：

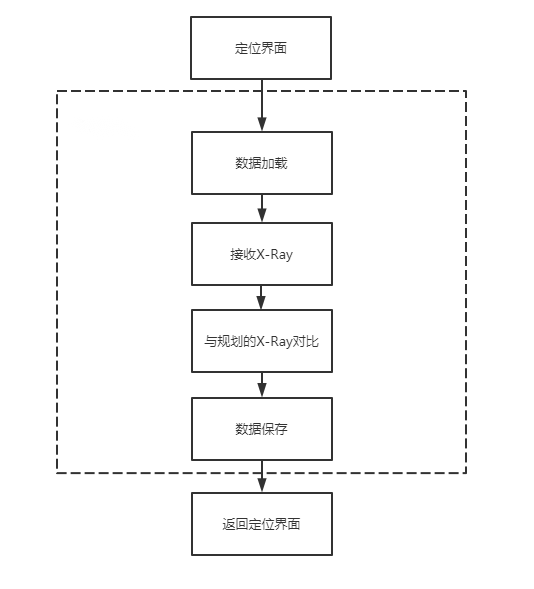
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | 接收到的新的dcm数据 | 无 | (FMODEL(Orth2DModel)->feedback\_dcm | 不为空 | 自动保存 | 将发送过来的dcm保存到本地 |

## 算法

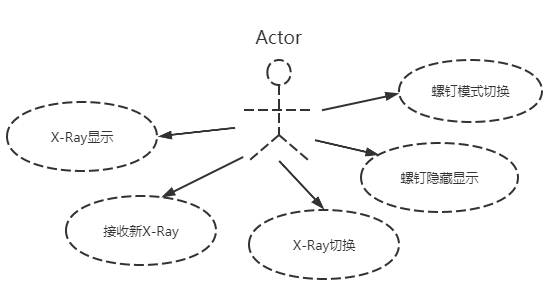
反馈不包含算法。

## 程序逻辑

反馈是规划定位系统流程的一个可选步骤，在定位界面点击评估按钮进入到反馈界面。分别用反馈在软件内的流程图、用例图来展示本功能的程序逻辑。



反馈在软件内的流程图



反馈用例图

## 接口

反馈包含多个主要类。其中重要接口详细说明如下：

/\*

\*功能描述: 切换视图1上一张X-Ray

\*参 数: 无

\*返 回 值: 无

\*/

void on\_up\_btn1\_clicked();

/\*

\*功能描述: 切换视图2上一张X-Ray

\*参 数: 无

\*返 回 值: 无

\*/

void on\_up\_btn2\_clicked();

/\*

\*功能描述: 切换视图1下一张X-Ray

\*参 数: 无

\*返 回 值: 无

\*/

void on\_down\_btn2\_clicked();

/\*

\*功能描述: 切换视图2下一张X-Ray

\*参 数: 无

\*返 回 值: 无

\*/

void on\_down\_btn2\_clicked();

/\*

\*功能描述: 视图1螺钉样式切换

\*参 数: 无

\*返 回 值: 无

\*/

void on\_dm\_btn\_clicked();

/\*

\*功能描述: 视图1螺钉隐藏显示切换

\*参 数: 无

\*返 回 值: 无

\*/

void on\_eye\_btn\_clicked();

/\*

\*功能描述: 恢复数据

\*参 数: 无

\*返 回 值: 无

\*/

void RecoverData();

/\*

\*功能描述: 处理DCM

\*参 数:

\*\*参 数0: dcm保存路径

\*返 回 值: 无

\*/

void DcmProcess(QString path);

## 存储分配

反馈功能本地文件存储内容：

X-Ray保存到当前病例目录下： "./Feedback/"目录。

二维反馈内存存储内容：

反馈的X-Ray数据: (FMODEL(Orth2DModel)->feedback\_dcm；

## 限制条件

1. 点击关闭按钮离开功能；
2. 接收X-Ray前要建立通讯连接；
3. 接收X-Ray有空间存储限制；

## 测试要点

1. 查看接收X-Ray是否显示；
2. 验证空间限制是否起作用；
3. 查看切换X-Ray是否显示正确，X-Ray顺序是否正确，X-Ray数量指示标签是否在切换中显示正常；

# 配置管理模块

## 模块描述

配置管理模块通过管理软件配置文件实现对软件配置参数的管理，与其它功能耦合度较小。

## 功能

软件处于运行状态，点击软件顶部的设置按钮，系统会弹出配置管理模块界面。配置管理模块主要分为配置文件设置、网络设置、系统设置、日志导出以及添加账户功能，详细功能如下：

1. 设置界面可以导入导出工具包参数和C臂机参数，可以进行删除和覆盖。
2. 工具包参数里面包含一些主要参数：

C\_arm\_points\_：C臂机配准板在三坐标仪标定后点的坐标位置。

registpara\_robotic\_arm\_：计算图像矩阵时用的配准初始化参数。

robot\_points\_：机械臂配准板在三坐标仪标定后点的坐标位置。

channel\_down\_：利用设备标定得到的通道数据终点。

channel\_down\_：利用设备标定得到的通道数据起点。

flange2robot\_tracker\_：机械臂法兰到track的映射矩阵。

1. 网络设置可以显示当前主机的IP地址、AEtitle以及端口号，这些参数用于C臂机通信，保证图像传输。
2. 系统设置提供各种系统选项设置，可以修改的选项有：医院名称修改、提示选项、定位流程以及屏保时间设置，机械臂和NDI的ip等一系列参数；
3. 图像小球识别算法参数、在算法文件中，主要有以下参数：

"min\_threshold\_": 二值化的起始阈值。

"max\_threshold\_": 二值化的终止阈值。

"min\_area\_": 斑点的最小面积。

"max\_area\_": 斑点的最大面积。

"threshold\_step\_": 二值化的阈值步长。 "min\_repeatability\_": 重复的最小次数 ，只有属于灰度图像斑点的那些二值图像斑点数量大于该值时，该灰度图像斑点才被认为是特征点。

"min\_circularity\_": 斑点的最小圆度。

"min\_convexity\_": 斑点的最小凸度。

"min\_inertia\_ratio\_": 斑点的最小惯性率。

"threshold\_": CT显示阈值。

"adjust\_step\_": 移动步长。

1. 日志导出详见20日志管理模块;
2. 账户添加提供软件账户添加功能。添加后的数据存放在指定json文件中，对应登录时读取json，能准确显示新增账户。



配置管理模块功能IPO图

## 性能

无。

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 系统配置 | 无 | json | 符合json格式 | 自动输入 | 系统配置保存的数据 |

## 输出项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | 系统配置 | 无 | json | 符合json格式 | 鼠标点击保存 | 保存系统配置的数据 |

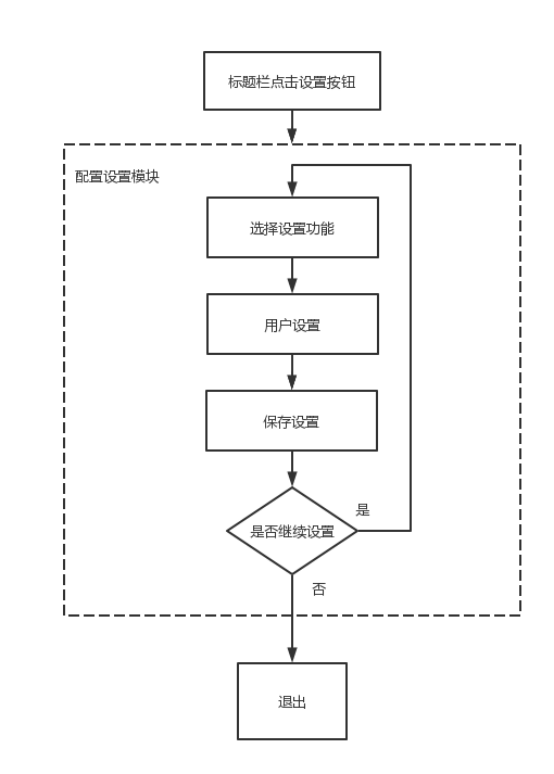
## 算法

无。

## 程序逻辑

配置管理模块位于软件顶部工具栏，软件登录后点击软件设置按钮后开启。

用配置管理模块在软件内的流程图来展示本功能的程序逻辑。



配置设置模块在软件内流程图

## 接口

配置设置模块主要包含SettingWidget类，其中重要接口详细说明如下：

/\*

\*功能描述: 配置设置模块

\*/

class SettingWidget;

/\*

\*功能描述: 读取网络设置界面配置

\*/

void ReadNetPageConfig();

/\*

\*功能描述: 读取系统设置界面配置

\*/

void ReadConfigPageConfig();

/\*

\*功能描述: 读取工具包界面配置

\*/

void ReadToolPageConfig();

/\*

\*功能描述: 读取C臂机界面配置

\*/

void ReadCarmPageConfig();

/\*

\*功能描述: 导入工具包参数按钮按下触发槽

\*/

void ImportToolConfig();

/\*

\*功能描述: 导出工具包参数按钮按下触发槽

\*/

void ExportToolConfig();

/\*

\*功能描述: 添加账户槽函数

\*/

void AccountButtonClicked();

/\*

\*功能描述: 导出日历按钮按下槽函数

\*/

void OutputButtonClicked();

## 存储分配

配置设置模块内存存储屏保时间：

FMODEL(MainModel)->screen\_protection\_

配置设置模块本地存储：

工具包、C臂机配置文件分别在“config/tool\_package/”和“config/c\_arm\_package”目录下；

系统配置文件存储在"config/"目录下。

## 限制条件

1. 管理员账户只能添加软件账户，其他模块无法进入；
2. 操作员账户无法进入配置设置模块。

## 测试要点

1. 登录不同账户查看对应是否可以打开；
2. 是否可以修改设置配置文件；
3. 恶意修改配置文件是否可以进行数据恢复；
4. 错误的工具包/C臂机配置文件是否可以导入；
5. 切换功能是否流畅；
6. 网络配置是否正确获取；
7. 系统设置保存是否生效。

# 设备自检模块

## 模块描述

1. 本模块是针对手术过程中使用的各种手术设备进行精度自检，与其他功能无耦合。
2. 设备自检功能在软件中属于常驻内存、可以重入，不属于顺序处理且代码没有覆盖要求。

## 功能

软件处于运行状态，点击软件顶部工具栏的设备自检按钮，系统会弹出设备自检界面。设备自检模块分为探针检测、双目相机检测、配准板检测、骨钻标定和套筒标定，其详细功能如下：

1. 探针检测有动画提示将探针放置在标定器对应位置，按下“确认”按钮进行检测。检测到的结果会反馈在界面上；
2. 双目相机检测有动画提示将探针尖端放置在标尺A处，点击“确定”后记录数据，动画提示将探针尖端放置在标尺B处，点击“确定”后记录数据并反馈结果；
3. 配准板检测分为C臂机配准板检测和机械臂配准板检测：

C臂机配准板检测：有动画提示将示踪器安装在C臂机配准板上，并放置在双目相机的可视范围内，再将探针的针尖放置在配准板的一个位置上，点击“确定”按钮后记录数据，动画跳转至下一个记录的位置，所有数据记录完成后反馈结果在界面上；

机械臂配准板检测功能同C臂机配准板检测；

1. 骨钻标定有动画提示将骨钻放入标定器对应位置，点击“确定”按钮后，对骨钻进行标定，完成后结果反馈在界面上；
2. 套筒标定有动画提示将套筒和探针放置在指定位置，点击“确认”按钮后，对套筒进行标定，完成后将结果反馈在界面上；
3. 双目相机识别到待检测设备会有显示提示，只有当被检测设备同时被双目相机识别到时，才可以按下“确认”按钮。

## 性能

无。

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 标定示踪器 | 无 | NDIToolData | 不为空 | 自动输入 | 双目相机获取的标定器数据 |
| **2** | 探针示踪器 | 无 | NDIToolData | 不为空 | 自动输入 | 双目相机获取的探针数据 |
| **3** | 机械臂示踪器 | 无 | NDIToolData | 不为空 | 自动输入 | 双目相机获取的机械臂配准板数据 |
| **4** | C臂机示踪器 | 无 | NDIToolData | 不为空 | 自动输入 | 双目相机获取的C臂机配准板数据 |
| **5** | 套筒示踪器 | 无 | NDIToolData | 不为空 | 自动输入 | 双目相机获取的套筒数据 |
| **6** | 骨钻示踪器 | 无 | NDIToolData | 不为空 | 自动输入 | 双目相机获取的骨钻数据 |

## 输出项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | 探针检测数据 | 无 | Json | 不为空 | 点击确认按钮 | 更新工具包中探针结果 |
| **2** | 双目相机检测数据 | 无 | Json | 不为空 | 点击确认按钮 | 双目相机检测结果 |
| **3** | 套筒标定数据 | 无 | Json | 不为空 | 点击确认按钮 | 套筒标定结果 |
| **4** | 骨钻标定数据 | 无 | Json | 不为空 | 点击确认按钮 | 骨钻标定结果 |
| **5** | 机械臂检测数据 | 无 | Json | 不为空 | 点击确认按钮 | 机械臂检测结果 |
| **6** | C臂机检测数据 | 无 | Json | 不为空 | 点击确认按钮 | C臂机检测结果 |

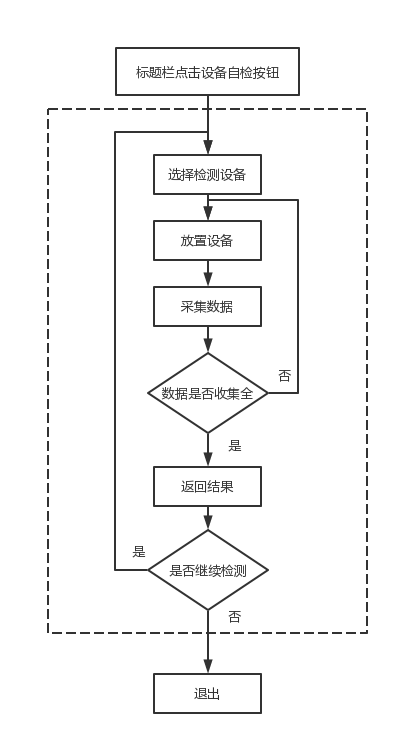
## 算法

无。

## 程序逻辑

设备自检模块位于软件顶部工具栏，软件登录后点击设备自检按钮后开启。

用设备自检模块在软件内的流程图来展示本功能的程序逻辑。



设备自检在软件内流程图

## 接口

设备检测模块主要包含DeviceDetectionWidget类，其中重要接口详细说明如下：

19.8.1 DeviceDetectionWidget

/\*\*

\* @brief The DeviceDetectionWidget class

\*

\*/

class DeviceDetectionWidget;

/\*

\*功能描述: 探针检测确认按钮槽

\*/

void StyletOkButtonClicked();

/\*

\*功能描述: 双目相机检测确认按钮槽

\*/

void CameraOkButtonClicked();

/\*

\*功能描述: 套筒标定确认按钮槽

\*/

void SleeveOkButtonClicked();

/\*

\*功能描述: 骨钻标定确认按钮槽

\*/

void BoneOkButtonClicked();

/\*

\*功能描述: C臂机配准板检测确认按钮槽

\*/

void CArmOkButtonClicked();

/\*

\*功能描述: 机械臂配准板检测确认按钮槽

\*/

void RobotArmOkButtonClicked();

## 存储分配

设备自检管理模块没有内存存储内容，日志管理模块本地存储内容：

1. 将检测的设备数据同步更新至对应的本地json文件中。

## 限制条件

1. 进入模块需要登录软件

## 测试要点

1. 设备能否检测到；
2. 设备检测中途断开设备连接是否能正确更新断开状态；
3. 反馈信息是否正确；
4. 本地文件数据是否成功更新。

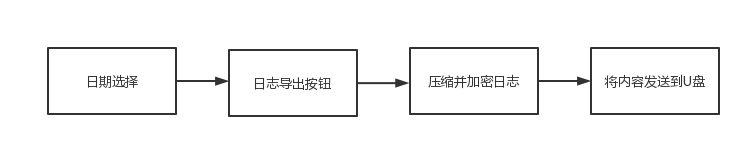
# 日志管理模块

## 模块描述

1. 日志管理模块用来将记录的日志文件压缩并加密导出到U盘，保证了软件网络安全的健康数据存储保密性能力；
2. 日志管理模块在软件中属于常驻内存、可重入，不属于顺序处理。本功能与其它功能耦合度较小。

## 功能

1. 软件成功登录后在设置界面提供导出日志功能按钮；
2. 日志管理模块可以判断U盘是否存在，用进度条展示导出进度。对导出到U盘的日志压缩包进行加密处理，导出过程中可以手动取消导出操作；
3. 软件未登录时无法提供导出功能。



日志管理模块功能IPO图

## 性能

无。

## 输入项目

日志管理模块输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | 起始日期 | 无 | QDate | 不为空 | 鼠标选择 | 选择导出日志的起始日期 |
| 2 | 截止日期 | 无 | QDate | 不为空 | 鼠标选择 | 选择导出日志的截止日期 |
| 3 | 本地日志 | 无 | .txt | 不为空 | 自动输入 | 本地已保存日志 |
| 4 | U盘名称 | 无 | QString | 不为空 | 鼠标选择 | 选择导出的U盘 |

## 输出项目

日志管理模块输入项目如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| 1 | 加密日志文件 | 无 | Zip | 不为空 | 自动保存 | 导出到U盘的加密日志文件 |

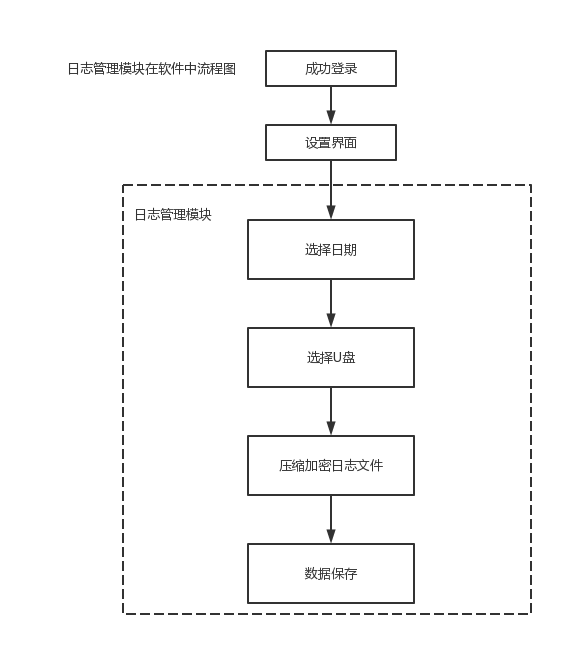
## 算法

无。

## 程序逻辑

日志管理模块独立于其他所有功能随软件登录而开启，随软件关闭而结束。

用日志管理模块在软件内的流程图来展示本功能的程序逻辑。



日志管理模块在软件内的流程图

## 接口

日志导出功能包含LoggerConfigTool主要类。

20.8.1

/\*\*

\* @brief The LoggerConfigTool class

\* 日志管理模块主要类

\*/

class LoggerConfigTool;

/\*\*

\* @brief OutputLogFiles

\* 导出日志

\*/

static bool OutputLogFiles(QDate start\_date, QDate end\_date, QString out\_path);

## 存储分配

日志管理模块内存存储内容：

日志文件列表(全局唯一)：LoggerConfigTool::instance()

日志管理模块本地存储内容：

日志导出文件保存在选择U盘根目录下，文件名为log和当前导出时间组合的zip压缩包。

## 限制条件

1. 随软件登录而开启，随软件关闭而结束；
2. 本地必须选择起始和截止日期；
3. 必须插入并手动选择U盘。

## 测试要点

1. 是否正常识别单个或者多个U盘；
2. 导出过程能否取消导出；
3. 是否可以正常导出；
4. 导出日志文件是否加密；
5. 导出时间是否合法。

# 双目相机模块

## 模块描述

1. 双目相机模块主要是展示识别到的示踪器在双目相机下的位置；
2. 本功能与其他模块的耦合较小，用户可以通过点击双目相机按钮，进入双目相机界面。

## 功能

1. 双目相机的手动重连；
2. 依据相机坐标系，建立Z轴视图与Y轴视图显示相机工作区域；
3. 识别到的示踪器根据空间位置显示在两个视图中对应位置；
4. 能够获取注册示踪器的位置、姿态与误差

## 性能

无

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 设备名称 | 无 | String | 有效的设备名称 | 自动输入 | 无 |

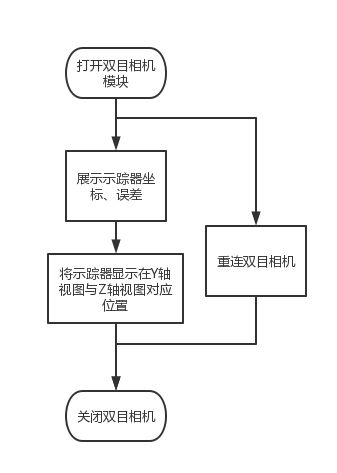
## 输出项目

|  |  |  |
| --- | --- | --- |
| **名称** | **标识** | **数据类型** |
| 示踪器在双目相机下的坐标 | tool\_loc\_ | Double数组 |
| 误差 | tool\_error\_ | Double |

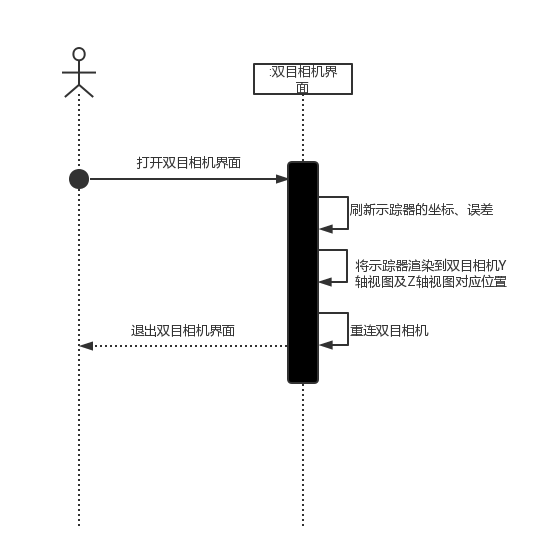
## 算法

无

## 程序逻辑



双目相机模块在软件内的流程图



双目相机模块时序图

## 接口

双目相机模块包含NDIViewWidget、NDIMapping两个主要类。其中重要接口详细说明如下：

21.8.1. NDIViewWidget

/\*

\* 功能描述: ndi 数据接收与处理

\*/

class NDIViewWidget;

/\*

\* 初始化示踪器

\*/

void InitWidget();

/\*

\* 功能描述：更新ndi示踪器数据

\*/

void STUpdate() override;

/\*

\* 功能描述：连接ndi

\*/

void Connect();

/\*

\* 功能描述：绘制示踪器

\* 参数：示踪器类型

\*/

void PaintSzq(NDITOOL tool);

### 18.8.1.NDIMapping

/\*

\* 功能描述: 在界面上绘制示踪器

\*/

class NDIMapping ;

/\*

\* 功能描述：根据示踪器的坐标位置在界面上绘制示踪器

\* 参数1：示踪器类型

\* 参数2：示踪器坐标

\*/

void PaintTracer(NDITOOL type, double\* position);

/\*

\* 功能描述：在界面上清除某一类型的示踪器

\* 参数：示踪器类型

\*/

void ClearTracer(NDITOOL type);

/\*

\* 功能描述：设置当前界面对应的ndi方向

\* 参数：ndi方向

\*/

void SetDirection(Direction direction);

/\*

\* 功能描述：设置当前界面的背景

\*/

void SetBackgrondImage(QString path);

## 存储分配

无

## 限制条件

1. 双目相机相关工具包已选择。

## 测试要点

1. 示踪器位置能否正常显示；
2. 示踪器状态是否正常；
3. 双目相机是否能重连。

# 机械臂控制模块

## 模块描述

1. 机械臂控制模块是一个弹窗，对机械臂关节角，位姿展示，并可对各个关节角和位姿进行调节。
2. 收纳位，左展开位，右展开位功能按钮可将机械臂移动到固定的位置，自由拖动按钮可将机械臂移动到任意位置。
3. 自定位位姿保存功能可将特定的位姿保存，需要时执行移动到特定的位姿。
4. 状态提示框对机械臂启动状态和异常状态展示。
5. 提供了控制机械臂连接，断开，重启，关机，解除停机的功能按钮。

## 功能

1. 点击功能菜单的机械臂图标，可弹出机械臂控制模块弹窗，左侧展示机械臂模型表示机械臂当前实际位姿，右侧显示机械臂姿态，关节角的当前数据。
2. 长按关节角控制的“+”“-”按钮，可以移动机械臂各个关节角。长按姿态控制的“+”，“-”按钮，可以改变机械臂的位姿。
3. 长按自由拖动按钮，可以拖动机械臂到合适的位置。长按收纳位，左展开位，右展开位按钮，可分别移动机械臂到指定位置。
4. 点击保存当前位置按钮，可以保存当前机械臂的各个关节角的角度，同时左侧下拉菜单中增加一个选项，右击下拉菜单，可对保存的位置重命名。点击执行按钮，可以移动机械臂到保存的位姿。
5. 分别点击连接，断开，重启，关机按钮，可以控制机械臂的连接，断开，重启，关机。机械臂保护性停止，可以通过点击解除停机按钮恢复到正常状态。

## 性能

无

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | 自定位关节角 | actual\_joint\_list\_ | Double | -330-330 | 输入框 |  |
| 2 | 自定位名称 | tcp\_pose\_name\_ | QString | 20字节中英文数字混合输入 | 输入框 |  |

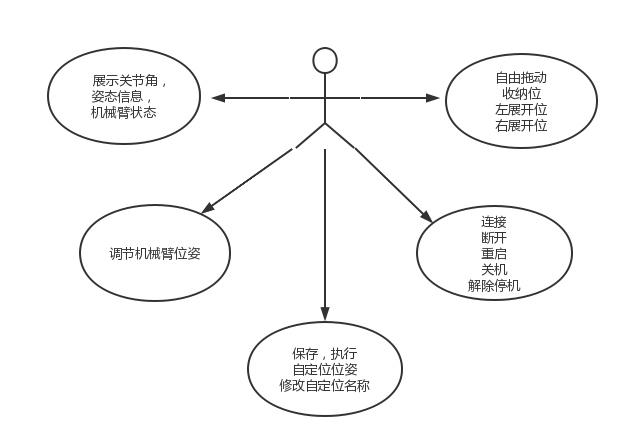
## 输出项目

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **标识** | **数据类型** | **有效范围** |
| 关节角 | joint\_angle\_vec | std::vector<double> | -330-330 |
| 位姿 | tcp\_pose\_vec | std::vector<double> |  |

## 算法

无

## 程序逻辑



## 接口

机械臂控制模块主要包含RobotParaAdjustWidget，RobotSimulationWidget，RobotArmViewWidget类。主要的接口如下:

### RobtParaAdjustWidget

/\*

功能描述:显示关节角，位姿的数值

\* 参数:

\* \*参数0:关节角或位姿的数值

\* 返回值:无

\*/

void SetRobotParaValue(double robot\_value);

/\*

功能描述:机械臂异常状态关节角，位姿的数值显示

\* 参数:无

\* 返回值:无

\*/

void SetInvalidParaValue();

/\*

功能描述:关节角，位姿反向移动

\* 参数:无

\* 返回值:无

\*/

void on\_down\_tool\_button\_pressed();

/\*

功能描述:关节角，位姿反向移动停止

\* 参数:无

\* 返回值:无

\*/

void on\_down\_tool\_button\_released();

/\*

功能描述:关节角，位姿正向移动

\* 参数:无

\* 返回值:无

\*/

void on\_up\_tool\_button\_pressed();

/\*

功能描述:关节角，位姿正向移动停止

\* 参数:无

\* 返回值:无

\*/

void on\_up\_tool\_button\_released();

### RobotSimulationWidget

/\*

功能描述:创建坐标系

\* 参数:无

\* 返回值:vtkAxesActor

\*/

vtkSmartPointer<vtkAxesActor> CreateAxesActor();

/\*

功能描述:创建水平面Actor

\* 参数:无

\* 返回值:vtkActor

\*/

vtkSmartPointer<vtkActor> CreatePlaneActor();

/\*

功能描述:机械臂模型装配体刷新

\* 参数:无

\*\*参数0:六个轴的关节角

\* 返回值:无

\*/

void FreshAssembly(std::vector<double>& joint\_angle\_vec);

### RobotArmViewWidget

/\*

功能描述:开始自由拖动

\* 参数:无

\* 返回值:无

\*/

void on\_freedrag\_push\_button\_pressed();

/\*

功能描述:自由拖动停止

\* 参数:无

\* 返回值:无

\*/

void on\_freedrag\_push\_button\_released();

/\*

功能描述:机械臂连接

\* 参数:无

\* 返回值:无

\*/

void on\_connect\_push\_button\_clicked();

/\*

功能描述:机械臂断开

\* 参数:无

\* 返回值:无

\*/

void on\_disconnect\_push\_button\_clicked();

/\*

功能描述:机械臂重启

\* 参数:无

\* 返回值:无

\*/

void on\_reboot\_push\_button\_clicked();

/\*

功能描述:机械臂关机

\* 参数:无

\* 返回值:无

\*/

void on\_halt\_push\_button\_clicked();

/\*

功能描述:机械臂解除停机

\* 参数:无

\* 返回值:无

\*/

void on\_unlockstop\_push\_button\_clicked();

/\*

功能描述:保存自定位位姿

\* 参数:无

\* 返回值:无

\*/

void on\_save\_pos\_push\_button\_clicked();

/\*

功能描述:执行自定位位姿

\* 参数:无

\* 返回值:无

\*/

void on\_exec\_pos\_push\_button\_pressed();

/\*

功能描述:执行自定位位姿停止

\* 参数:无

\* 返回值:无

\*/

void on\_exec\_pos\_push\_button\_released();

/\*

功能描述:自定位位姿重命名

\* 参数:无

\* 返回值:无

\*/

void SlotRenameTcpPose();

## 存储分配

自定位位姿保存在config文件夹下的system\_config.json中

## 限制条件

1. 关节角范围在-330-330
2. 自定位名称为20个字节中英文，数字

## 测试要点

1. 关节角，位姿是否显示正常
2. 机械臂上电状态，安全异常状态是否显示正常
3. 自由拖动，左右展开位，收纳位功能测试
4. 自定位的保存，执行，重命名功能测试
5. 机械臂连接，断开，重启，关机，解除停机功能测试

# C臂机模块

## 模块描述

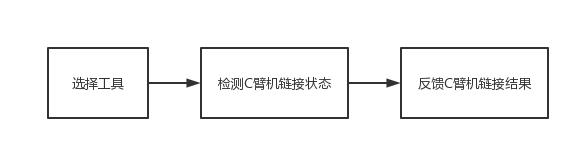
1. C臂机模块功能是本软件的重要组成部分，主要用于检测外部C臂机同软件的连接是否正常，与其他功能耦合度小；
2. C臂机模块功能在软件中属于常驻内存、不可重入，不属于顺序处理，且代码没有覆盖要求。

## 功能

C臂机模块功能独立于软件顺序流程，登录成功后客户端后台静默运行。

C臂机模块功能详细功能：

1. 软件登录后，选择患者数据以及对应的C臂机配置参数后，启动C臂机模块功能，并在主界面标题栏显示与C臂机的连接状态，连通显示绿色，未连通显示灰色。
2. 支持用户切换C臂机参数。



C臂机模块功能IPO图

## 性能

1. 收到并解析数据响应时间在1s内；
2. 发送数据响应时间在1s内。

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| 1 | C臂机Ip地址 | 无 | QString | 符合IPV4协议 | 配置参数读取 | 选择的C臂机  Ip地址 |

## 输出项目

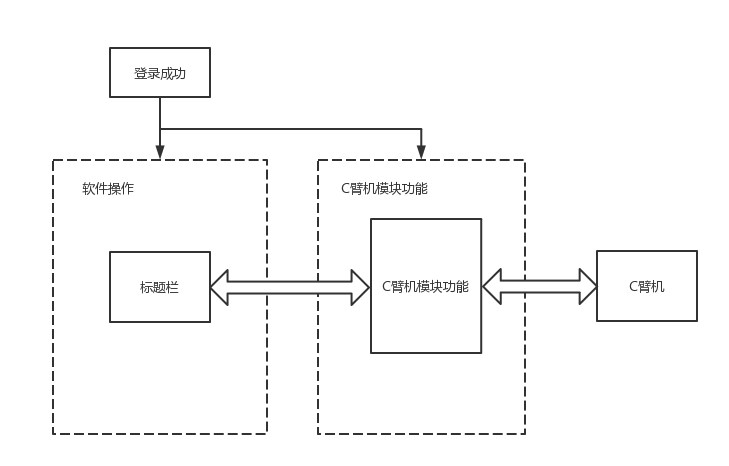
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| 1 | C臂机连接状态 | 无 | Bool | 不为空 | 状态栏显示 | 返回C臂机  是否连接 |

## 算法

无。

## 程序逻辑

C臂机模块功能独立于其他功能，随C臂机参数选择完成后启动。



C臂机模块功能流程图

## 接口

C臂机模块主要包含CarmThread类，其中重要接口详细说明如下：

/\*

\*功能描述: 返回C臂机连接状态

\*/

bool *CarmConnectState() ;*

/\*

\*功能描述: 设置C臂机IP

\*/

void *SetCarmIPaddr(QString ip)*;

/\*

\*功能描述: 结束线程

\*/

void *CarmEndTrack()*;

## 存储分配

无。

## 限制条件

1. 必须选择C臂机配置参数

## 测试要点

1. 正确连接C臂机查看是否显示已连接；
2. 修改配置参数查看C臂机连接状态是否正确；
3. 能否正常退出。

# 系统功能模块

## 模块描述

1. 系统功能模块包括设备状态显示、手术信息显示、软件信息、截图功能、设备自检模块、软件配置设置以及系统退出；
2. 系统功能模块在软件中属于常驻内存、不可重入，不属于顺序处理且代码无覆盖要求。本模块与其他功能无耦合。

## 功能

系统登录后自动开启，位于软件顶部显示，随软件关闭而关闭。

系统功能模块详细功能：

1. 正确显示软件信息和公司名称UI，布局设计、风格合理；
2. 显示当前手术信息(患者姓名和手术器械)；
3. 可以正常打开机械臂控制模块，加载机械臂模型数据在三维视图中显示，模型关节姿态与机械臂保持一致；
4. 可以正常打开双目相机模块，识别到的示踪器根据空间位置显示在两个视图中对应位置，依据相机坐标系，建立Z轴视图与Y轴视图显示相机工作区域；
5. 打开系统设置配置管理，设置相关参数，保存后软件能根据修改生效；
6. 打开软件信息界面，准确显示软件的相关信息；
7. 提供截图功能，利用QPropertyAnimation方法，实现属性来控制动画，点击截图后，屏幕图像保存到指定文件夹；
8. 语音播报相关设置，利用QTextToSpeech类提供了对文本到语音引擎的方便访问。可以设置语音声音和速度。
9. 设置不同颜色的界面样式，可供选择。
10. 用户具体不同权限，且用户无法进入后台操作系统，不同权限对于不同功能。
11. 程序设定操作时间，超出设置时间无操作，系统进入锁屏，输入密码后才能继续使用.
12. 软件操作过程中，如果触发错误，弹框提示相关信息。
13. 软件依赖文件进行格式与路径检查，如果触发错误，弹框提示相关信息。
14. 软件启动后，写入系统注册表，实现重启后软件自启动。
15. 软件关闭、重启以及锁屏功能，点击关闭按钮，弹窗显示关机、锁屏、重启选项，点击关机按钮，系统设备自动关机，点击锁屏按钮，系统进入锁屏，点击重启按钮，系统设备重启。



系统功能模块功能IPO图

## 性能

无。

## 输入项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输入方式** | **备注** |
| **1** | 患者信息 | 无 | QString | 不为空 | 自动输入 | 患者姓名信息 |
| **2** | 手术器械 | 无 | QString | 不为空 | 自动输入 | 本次手术所选器械信息 |

## 输出项目

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **序号** | **名称** | **标识** | **类型** | **有效范围** | **输出方式** | **备注** |
| **1** | 截图 | 无 | .jpg | 不为空 | 本地文件 | 当前屏幕截图 |

## 算法

无。

## 程序逻辑

系统功能模块在软件登录成功后开启，随软件关闭而关闭。

可以根据模块按钮提示打开对应模块功能。

## 接口

系统功能模块主要包含TitlebarWidget，其中重要接口详细说明如下：

/\*

\*功能描述: 构造函数

\*/

class TitlebarWidget;

/\*

\*功能描述: 打开设备检测界面

\*/

void on\_cali\_clicked();

/\*

\*功能描述: 打开ndi界面

\*/

void on\_ndi\_clicked();

/\*

\*功能描述: 打开机械臂界面

\*/

void on\_arm\_clicked();

/\*

\*功能描述: 打开设置界面

\*/

void on\_setting\_clicked();

/\*

\*功能描述: 关机按钮

\*/

void on\_exit\_clicked();

/\*

\*功能描述: 打开关于界面

\*/

void on\_about\_dialog\_clicked();

/\*

\*功能描述: 截图

\*/

void GetScreenShots();

## 存储分配

截图保存至本地目录中：

未选择患者信息时截图保存至“./data/screen\_shots”文件夹下，选择患者信息后，保存在对应患者信息目录下“screen\_shots”文件夹中。

## 限制条件

1. 必须登录软件

## 测试要点

1. 能否正确打开对应界面功能；
2. 软件信息是否显示完整；
3. 手术器械状态显示是否正确；
4. 手术信息显示是否正确；
5. 截图保存目录是否正确；
6. 能否正常关闭、重启软件。

# 其他要求

1. 界面有合理的布局设计、风格。
2. 标题栏要显示公司logo、系统名称、版本号。
3. 子窗口通过按钮或状态触发，子窗口为模态窗口。
4. 窗口开启关闭保持对数据的更新与保存，软件可以根据数据恢复窗口操作。
5. 软件操作过程中，如果触发错误，弹框提示相关信息。
6. 软件依赖文件进行格式与路径检查，如果触发错误，弹框提示相关信息。
7. 关键流程操作提示相关信息，引导用户。
8. 软件运行具有稳定性、可靠性、高效性。
9. 软件使用具有易用性和界面友好。
10. 软件数据具有安全性和隐私保护。